



Techniques of Water-Resources Investigations of the United States Geological Survey

Chapter A6

A COUPLED SURFACE-WATER AND GROUND- WATER FLOW MODEL (MODBRANCH) FOR SIMULATION OF STREAM-AQUIFER INTERACTION

By Eric D. Swain and Eliezer J. Wexler

Book 6
MODELING TECHNIQUES

U.S. DEPARTMENT OF THE INTERIOR
BRUCE BABBITT, *Secretary*

U.S. GEOLOGICAL SURVEY
Gordon P. Eaton, *Director*

Any use of trade, product, or firm names in this publication is for descriptive purposes only
and does not imply endorsement by the U.S. Government

UNITED STATES GOVERNMENT PRINTING OFFICE, WASHINGTON : 1996

For sale by U.S. Geological Survey, Information Services
Box 25286, Federal Center, Denver, CO 80225

PREFACE

The series of manuals on techniques describes procedures for planning and executing specialized work in water-resources investigations. The material is grouped under major subject headings called "Books" and further subdivided into sections and chapters. Section A of Book 6 is on ground-water models.

This chapter documents the theory and application of a new coupled ground-water and surface-water model that was developed by combining the USGS models MODFLOW and BRANCH. The interfacing code is referred to as MODBRANCH. Although BRANCH is modified to act as a subroutine or module of MODFLOW, the input data sets for both models are very similar to their original form. Only model changes implemented for the coupling are documented in this chapter. For specifics on each model, users are advised to refer to the Techniques of Water-Resources Investigations chapters on MODFLOW and BRANCH.

TECHNIQUES OF WATER-RESOURCES INVESTIGATIONS OF THE U.S. GEOLOGICAL SURVEY

The U.S. Geological Survey publishes a series of manuals describing procedures for planning and conducting specialized work in water-resources investigations. The manuals published to date are listed below and may be ordered by mail from the U.S. Geological Survey, Information Services, Box 25286, Federal Center, Denver, CO 80225 (an authorized agent of the Superintendent of Documents, Government Printing Office).

Prepayment is required. Remittance should be sent by check or money order payable to U.S. Geological Survey. Prices are not included in the listing below as they are subject to change. **Current prices can be obtained** by writing to the USGS address shown above. Prices include cost of domestic surface transportation. For transmittal outside the U.S.A. (except to Canada and Mexico) a surcharge of 25 percent of the net bill should be included to cover surface transportation. When ordering any of these publications, please give the title, book number, chapter number, and "U.S. Geological Survey Techniques of Water-Resources Investigations."

- TWRI 1-D1. Water temperature—influential factors, field measurement, and data presentation, by H.H. Stevens, Jr., J.F. Ficke, and G.F. Smoot. 1975. 65 pages.
- TWRI 1-D2. Guidelines for collection and field analysis of ground-water samples for selected unstable constituents, by W.W. Wood. 1976. 24 pages.
- TWRI 2-D1. Application of surface geophysics to ground-water investigations, by A.A.R. Zohdy, G.P. Eaton, and D.R. Mabey. 1974. 116 pages.
- TWRI 2-D2. Application of seismic-refraction techniques to hydrologic studies, by F.P. Haeni. 1988. 86 pages.
- TWRI 2-E1. Application of borehole geophysics to water-resources investigations, by W.S. Keys and L.M. MacCary. 1971. 126 pages.
- TWRI 2-E2. Borehole geophysics applied to ground-water investigations, by W. Scott Keys. 1990. 150 pages.
- TWRI 2-F1. Application of drilling, coring, and sampling techniques to test holes and wells, by Eugene Shuter and Warren E. Teasdale. 1989. 97 pages.
- TWRI 3-A1. General field and office procedures for indirect discharge measurements, by M.A. Benson and Tate Dalrymple. 1967. 30 pages.
- TWRI 3-A2. Measurement of peak discharge by the slope-area method, by Tate Dalrymple and M.A. Benson. 1967. 12 pages.
- TWRI 3-A3. Measurement of peak discharge at culverts by indirect methods, by G.L. Bodhaine. 1968. 60 pages.
- TWRI 3-A4. Measurement of peak discharge at width contractions by indirect methods, by H.F. Matthai. 1967. 44 pages.
- TWRI 3-A5. Measurement of peak discharge at dams by indirect methods, by Harry Hulsing. 1967. 29 pages.
- TWRI 3-A6. General procedure for gaging streams, by R.W. Carter and Jacob Davidian. 1968. 13 pages.
- TWRI 3-A7. Stage measurement at gaging stations, by T.J. Buchanan and W.P. Somers. 1968. 28 pages.
- TWRI 3-A8. Discharge measurements at gaging stations, by T.J. Buchanan and W.P. Somers. 1969. 65 pages.
- TWRI 3-A9.¹ Measurement of time of travel in streams by dye tracing, by F.A. Kilpatrick and J.F. Wilson, Jr. 1989. 27 pages.
- TWRI 3-A10. Discharge ratings at gaging stations, by E.J. Kennedy. 1984. 59 pages.
- TWRI 3-A11. Measurement of discharge by the moving-boat method, by G.F. Smoot and C.E. Novak. 1969. 22 pages.
- TWRI 3-A12. Fluorometric procedures for dye tracing, Revised, by J.F. Wilson, Jr., E.D. Cobb, and F.A. Kilpatrick. 1986. 34 pages.
- TWRI 3-A13. Computation of continuous records of streamflow, by E.J. Kennedy. 1983. 53 pages.
- TWRI 3-A14. Use of flumes in measuring discharge, by F.A. Kilpatrick, and V.R. Schneider. 1983. 46 pages.
- TWRI 3-A15. Computation of water-surface profiles in open channels, by Jacob Davidian. 1984. 48 pages.
- TWRI 3-A16. Measurement of discharge using tracers, by F.A. Kilpatrick and E.D. Cobb. 1985. 52 pages.
- TWRI 3-A17. Acoustic velocity meter systems, by Antonius Laenen. 1985. 38 pages.
- TWRI 3-A18. Determination of stream reaeration coefficients by use of tracers, by F.A. Kilpatrick, R.E. Rathbun, N. Yotsukura, G.W. Parker, and L.L. DeLong. 1989. 52 pages.
- TWRI 3-A19. Levels at streamflow gaging stations, by E.J. Kennedy. 1990. 31 pages.
- TWRI 3-A20. Simulation of soluble waste transport and buildup in surface waters using tracers, by F.A. Kilpatrick. 1993. 38 pages.
- TWRI 3-A21. Stream-gaging cableways, by C. Russell Wagner. 1995. 56 pages.
- TWRI 3-B1. Aquifer-test design, observation, and data analysis, by R.W. Stallman. 1971. 26 pages.
- TWRI 3-B2.² Introduction to ground-water hydraulics, a programed text for self-instruction, by G.D. Bennett. 1976. 172 pages.

¹This manual is a revision of "Measurement of Time of Travel and Dispersion in Streams by Dye Tracing," by E.F. Hubbard, F.A. Kilpatrick, L.A. Martens, and J.F. Wilson, Jr., Book 3, Chapter A9, published in 1982.

²Spanish translation also available.

- TWRI 3-B3. Type curves for selected problems of flow to wells in confined aquifers, by J.E. Reed. 1980. 106 pages.
- TWRI 3-B4. Regression modeling of ground-water flow, by Richard L. Cooley and Richard L. Naff. 1990. 232 pages.
- TWRI 3-B4, Supplement 1. Regression modeling of ground-water flow—Modifications to the computer code for nonlinear regression solution of steady-state ground-water flow problems, by R.L. Cooley. 1993. 8 pages.
- TWRI 3-B5. Definition of boundary and initial conditions in the analysis of saturated ground-water flow systems—An introduction, by O. Lehn Franke, Thomas E. Reilly, and Gordon D. Bennett. 1987. 15 pages.
- TWRI 3-B6. The principle of superposition and its application in ground-water hydraulics, by Thomas E. Reilly, O. Lehn Franke, and Gordon D. Bennett. 1987. 28 pages.
- TWRI 3-B7. Analytical solutions for one-, two-, and three-dimensional solute transport in ground-water systems with uniform flow, by Eliezer J. Wexler. 1991. 193 pages.
- TWRI 3-C1. Fluvial sediment concepts, by H.P. Guy. 1970. 55 pages.
- TWRI 3-C2. Field methods of measurement of fluvial sediment, by H.P. Guy and V.W. Norman. 1970. 59 pages.
- TWRI 3-C3. Computation of fluvial-sediment discharge, by George Porterfield. 1972. 66 pages.
- TWRI 4-A1. Some statistical tools in hydrology, by H.C. Riggs. 1968. 39 pages.
- TWRI 4-A2. Frequency curves, by H.C. Riggs. 1968. 15 pages.
- TWRI 4-B1. Low-flow investigations, by H.C. Riggs. 1972. 18 pages.
- TWRI 4-B2. Storage analyses for water supply, by H.C. Riggs and C.H. Hardison. 1973. 20 pages.
- TWRI 4-B3. Regional analyses of streamflow characteristics, by H.C. Riggs. 1973. 15 pages.
- TWRI 4-D1. Computation of rate and volume of stream depletion by wells, by C.T. Jenkins. 1970. 17 pages.
- TWRI 5-A1. Methods for determination of inorganic substances in water and fluvial sediments, by Marvin J. Fishman and Linda C. Friedman, editors. 1989. 545 pages.
- TWRI 5-A2. Determination of minor elements in water by emission spectroscopy, by P.R. Barnett and E.C. Mallory, Jr. 1971. 31 pages.
- TWRI 5-A3.³ Methods for the determination of organic substances in water and fluvial sediments, edited by R.L. Wershaw, M.J. Fishman, R.R. Grabbe, and L.E. Lowe. 1987. 80 pages.
- TWRI 5-A4.⁴ Methods for collection and analysis of aquatic biological and microbiological samples, by L.J. Britton and P.E. Greeson, editors. 1989. 363 pages.
- TWRI 5-A5. Methods for determination of radioactive substances in water and fluvial sediments, by L.L. Thatcher, V.J. Janzer, and K.W. Edwards. 1977. 95 pages.
- TWRI 5-A6. Quality assurance practices for the chemical and biological analyses of water and fluvial sediments, by L.C. Friedman and D.E. Erdmann. 1982. 181 pages.
- TWRI 5-C1. Laboratory theory and methods for sediment analysis, by H.P. Guy. 1969. 58 pages.
- TWRI 6-A1. A modular three-dimensional finite-difference ground-water flow model, by Michael G. McDonald and Arlen W. Harbaugh. 1988. 586 pages.
- TWRI 6-A2. Documentation of a computer program to simulate aquifer-system compaction using the modular finite-difference ground-water flow model, by S.A. Leake and D.E. Prudic. 1991. 68 pages.
- TWRI 6-A3. A modular finite-element model (MODFE) for areal and axisymmetric ground-water-flow problems, Part 1: Model Description and User's Manual, by L.J. Torak. 1993. 136 pages.
- TWRI 6-A4. A modular finite-element model (MODFE) for areal and axisymmetric ground-water flow problems, Part 2: Derivation of finite-element equations and comparisons with analytical solutions, by R.L. Cooley. 1992. 108 pages.
- TWRI 6-A5. A modular finite-element model (MODFE) for areal and axisymmetric ground-water-flow problems, Part 3: Design philosophy and programming details, by L.J. Torak. 1993. 243 pages.
- TWRI 6-A6. A coupled surface-water and ground-water flow model (MODBRANCH) for simulation of stream-aquifer interaction. 1996. 125 pages.
- TWRI 7-C1. Finite difference model for aquifer simulation in two dimensions with results of numerical experiments, by P.C. Trescott, G.F. Pinder, and S.P. Larson. 1976. 116 pages.
- TWRI 7-C2. Computer model of two-dimensional solute transport and dispersion in ground water, by L.F. Konikow and J.D. Bredehoeft. 1978. 90 pages.
- TWRI 7-C3. A model for simulation of flow in singular and interconnected channels, by R.W. Schaffranek, R.A. Baltzer, and D.E. Goldberg. 1981. 110 pages.
- TWRI 8-A1. Methods of measuring water levels in deep wells, by M.S. Garber and F.C. Koopman. 1968. 23 pages.
- TWRI 8-A2. Installation and service manual for U.S. Geological Survey monometers, by J.D. Craig. 1983. 57 pages.
- TWRI 8-B2. Calibration and maintenance of vertical-axis type current meters, by G.F. Smoot and C.E. Novak. 1968. 15 pages.

³This manual is a revision of TWRI 5-A3, "Methods of Analysis of Organic Substances in Water," by Donald F. Goerlitz and Eugene Brown, published in 1972.

⁴This manual supersedes TWRI 5-A4, "Methods for collection and analysis of aquatic biological and microbiological samples," edited by P.E. Greeson and others, published in 1977.

CONTENTS

Definition of Symbols	IX
Abstract	1
Introduction	1
Purpose and Scope	2
Approach	2
Mathematical Formulation	3
Incorporation of Leakage	3
Drying and Rewetting of River Channels	7
Steady-State Simulation	8
Model Documentation	8
Modifications to Main Code	9
Data Entry	17
Module BRC1AL	18
Narrative	18
Program Listing for Module BRC1AL	19
List of Variables	24
Module BRC1RP	24
Narrative	24
Program Listing for Module BRC1RP	26
List of Variables	30
Module BRC1FM	30
Narrative	30
Program Listing for Module BRC1FM	32
List of Variables	37
Module BRC1BD	38
Narrative	38
Program Listing for Module BRC1BD	40
List of Variables	45
Model BRANCH'	46
Narrative	47
Program Listing for Model BRANCH'	52
List of Variables	72
Simulations of Stream-Aquifer Interaction	73
Schemes for Comparison	73
Problem 1—Floodwave Propagation with Bank Storage	73
Problem 2—Steady-State Simulation, Backwater, and Distribution of Flows at Junctions	74
Problem 3—Rewetting of Channel by Recharge Wells	77
Problem 4—Field Model of L-31N Canal	80
Results	81
Conclusions	81
References Cited	88
Appendix I—Sample MODBRANCH Input (Selected Parts)	90
Appendix II—Sample MODBRANCH Output (Selected Parts)	97

FIGURES

1-5.	Diagrams showing:	
	1. Computation of head differences in the BRANCH and MODFLOW models	3
	2. Iteration procedure between MODFLOW and BRANCH'	4
	3. Arrangement of MODFLOW model cells and BRANCH stream reaches	4
	4. Schematic of module calling sequence	4
	5. Theoretical channel configuration to allow for channel drying	7
6-10.	Flowcharts of:	
	6. Allocation module (BRC1AL)	18
	7. Data entry module (BRC1RP)	25
	8. Formulation module (BRC1FM)	30
	9. Budget module (BRC1BD)	38
	10. Modified BRANCH'	47
11.	Expanded flowchart of step 8b in BRANCH'	49
12.	Discharge hydrograph simulated by the MODBRANCH and four-point models	75
13.	Discharge hydrograph simulated by the MODBRANCH and Pinder models for a site 50,000 feet from upstream boundary	75
14.	Diagram showing aquifer and river layout for steady-state problem	76
15-18.	Ground-water head contours produced by:	
	15. MODBRANCH for symmetric recharge	76
	16. Stream package for symmetric recharge	77
	17. MODBRANCH for asymmetric recharge	77
	18. Stream package for asymmetric recharge	78
19.	Diagram showing aquifer, river, and well layout for drying and rewetting problem	78
20.	Stage hydrographs for drying and rewetting problem	79
21, 22.	Ground-water head contours at:	
	21. 6, 10, and 16 hours	80
	22. 16 hours without river	81
23-25.	Stage hydrographs for time intervals of 3, 6, and 12 minutes at a site:	
	23. At the stream junction	82
	24. 4,250 feet downstream from the stream junction	82
	25. 1,060 feet upstream from the stream junction	83
26.	Map showing location of L-31N canal test reach	84
27.	Diagram showing field instrumentation at L-31N canal test reach	85
28.	Model aquifer grid for the L-31N canal field problem	86
29, 30.	Hydrographs showing:	
	29. Measured and computed stage at L-31N canal at mile 1	87
	30. Measured and computed discharge at L-31N canal	87

TABLES

1.	Example listing of a modified main program	10
2.	Flows calculated in BRANCH' and stream package of MODFLOW for symmetrical and asymmetrical recharge	78
3.	Measured and model computed ground-water levels at the L-31N canal test site	83

METRIC CONVERSION FACTORS AND VERTICAL DATUM

	Multiply	By	To obtain
	foot (ft)	0.3048	meter
	mile (mi)	1.609	kilometer
	foot per second (ft/s)	0.3048	meter per second
	foot per day (ft/d)	0.3048	meter per day
	foot per year (ft/yr)	0.3048	meter per year
	cubic foot per second (ft ³ /s)	0.02832	cubic meter per second
	cubic foot per hour (ft ³ /hr)	0.02832	cubic meter per hour

Sea level: In this report “sea level” refers to the National Geodetic Vertical Datum of 1929 (NGVD of 1929)—a geodetic datum derived from a general adjustment of the first-order level nets of both the United States and Canada, formerly called Sea Level Datum of 1929.

DEFINITION OF SYMBOLS

A = Cross-sectional area of channel	t = Time
B = Channel topwidth	U = Coefficient in ground-water flow equation in MODFLOW
b' = Thickness of riverbed	U_a = Wind speed
C = Leakage coefficient = K'/b'	V = Coefficient in ground-water flow equation in MODFLOW
C_d = Water surface drag coefficient	W = Volumetric flux per unit volume
DCFM = Dry channel friction multiplier, multiplied by σ when channel is dry	x = Coordinate direction in MODFLOW
F = Inflow term in MODFLOW	y = Coordinate direction in MODFLOW
g = Gravitational acceleration	Z = Stage in channel
H_{COEF} = Coefficient in ground-water flow equation in MODFLOW	Z_{BOT} = Elevation of river bottom
h = Head in aquifer	z = Coordinate direction in MODFLOW
i, j, k = Spatial coordinates	α = Coefficient in continuity equation in BRANCH
K = Hydraulic conductivity of aquifer	β = Momentum coefficient
K' = Hydraulic conductivity of riverbed	δ = Right-hand side of continuity equation in BRANCH
k = Friction factor $(n/1.49)^2$ in inch-pound units or n^2 in metric units	γ = Coefficient in continuity equation in BRANCH
L = Longitudinal distance down channel	Θ = Weighting factor for spatial derivatives in BRANCH
m = Time level	ϵ = Right-hand side of momentum equation in BRANCH
NTSAQ = Number of BRANCH' time intervals in one MODFLOW time step	χ = Weighting factor for averaged quantities in BRANCH
n = Manning's frictional factor	λ = Coefficient in momentum equation in BRANCH
P = Head-dependent inflow term in MODFLOW	μ = Coefficient in momentum equation in BRANCH
Q = Flow rate in channel	ϕ = Angle between wind direction and channel orientation
q = Outflow per unit length of channel	σ = Coefficient in momentum equation in BRANCH containing friction term
r = Average hydraulic radius	ρ = Coefficient in momentum equation in BRANCH
R = Coefficient in ground-water flow equation in MODFLOW	ρ_a = Air density
RHS = Right-hand side of ground-water flow equation in MODFLOW	ρ_w = Water density
S_s = Specific storativity	ω = Coefficient in momentum equation in BRANCH
T_s = Starting time interval when BRANCH' is entered from MODFLOW	ζ = Coefficient in momentum equation in BRANCH
	ξ = Wind friction term = $C_d \rho_a / \rho_w$

A Coupled Surface-Water and Ground-Water Flow Model (MODBRANCH) for Simulation of Stream-Aquifer Interaction

By Eric D. Swain and Eliezer J. Wexler

ABSTRACT

Ground-water and surface-water flow models traditionally have been developed separately, with interaction between subsurface flow and streamflow either not simulated at all or accounted for by simple formulations. In areas with dynamic and hydraulically well-connected ground-water and surface-water systems, stream-aquifer interaction should be simulated using deterministic responses of both systems coupled at the stream-aquifer interface. Accordingly, a new coupled ground-water and surface-water model was developed by combining the U.S. Geological Survey models MODFLOW and BRANCH; the interfacing code is referred to as MODBRANCH. MODFLOW is the widely used modular three-dimensional, finite-difference ground-water model, and BRANCH is a one-dimensional numerical model commonly used to simulate unsteady flow in open-channel networks.

MODFLOW was originally written with the River package, which calculates leakage between the aquifer and stream, assuming that the stream's stage remains constant during one model stress period. A simple streamflow routing model has been added to MODFLOW, but is limited to steady flow in rectangular, prismatic channels. To overcome these limitations, the BRANCH model, which simulates unsteady, nonuniform flow by solving the St. Venant equations, was restructured and incorporated into MODFLOW. Terms that describe leakage between stream and aquifer as a function of streambed conductance and differences in aquifer and stream stage were added to the continuity equation in BRANCH. Thus, leakage between the aquifer and stream can be calculated separately in each model, or leakages calculated in BRANCH can be used in MODFLOW. Total mass in the coupled models is accounted for and conserved.

The BRANCH model calculates new stream stages for each time interval in a transient simulation based on upstream boundary conditions, stream properties, and initial estimates of aquifer heads. Next, aquifer heads are calculated in MODFLOW based on stream stages calculated by BRANCH, aquifer properties, and stresses. This process is repeated until convergence criteria are met for head and

stage. Because time steps used in ground-water modeling can be much longer than time intervals used in surface-water simulations, provision has been made for handling multiple BRANCH time intervals within one MODFLOW time step. An option was also added to BRANCH to allow the simulation of channel drying and rewetting. Testing of the coupled model was verified by using data from previous studies; by comparing results with output from a simpler, four-point implicit, open-channel flow model linked with MODFLOW; and by comparison to field studies of L-31N canal in southern Florida.

INTRODUCTION

Mathematical modeling has been developed to a high degree of sophistication in ground-water and surface-water disciplines. However, the interactions of these two systems generally have not been simulated at all or have been accounted for by less sophisticated formulations.

The processes and simulation of ground-water and surface-water interactions have interested researchers for many years. Pinder and Sauer (1971) coupled the unsteady river equations with the two-dimensional ground-water flow equations to study bank storage effects. Zitta and Wiggert (1971) and Morel-Seytoux (1975) incorporated bank storage into continuous streamflow simulation. Hall and Moench (1972) and Land (1977) used the convolution integral to account for river losses to bank storage. Faye and Mayer (1990) used the U.S. Geological Survey (USGS) MODFLOW three-dimensional ground-water flow model (McDonald and Harbaugh, 1988) with its River package to model stream-aquifer relations in the northern coastal plain of Georgia. However, a scheme that couples two widely accepted models to accurately simulate the ground-water and surface-water flows and their interaction has not been developed.

A strong interest in stream-aquifer relations developed in southern Florida because of the extensive canal network that is in close hydraulic contact with the surficial aquifer. Because of the high hydraulic conductivities in the surficial aquifer (Wilson, 1982), the two systems respond rapidly to

each other. This response makes the coupling of two dynamic models necessary to appropriately simulate transient ground-water or surface-water conditions.

The USGS, in cooperation with the South Florida Water Management District, began a study in October 1988 to develop a coupled ground-water and surface-water flow model. The USGS modular three-dimensional, finite-difference ground-water flow model, MODFLOW, was modified to interface with the USGS unsteady surface-water flow model, BRANCH (Swain and Wexler, 1991).

PURPOSE AND SCOPE

This report documents the theory and application of a new coupled ground-water and surface-water flow model that was developed by combining the USGS models, MODFLOW and BRANCH. The interfacing code is referred to as MODBRANCH. The coupled models were applied to four tests. Results of the coupled models were compared to results of previous studies. Also, the ability of the coupled models to simulate conditions that could not be simulated accurately with separate models or with models using less deterministic or empirical algorithms was tested.

APPROACH

The MODFLOW ground-water flow model contains two packages that account for leakage to and from rivers and canals. The River package allows rivers to be represented with a stage fixed during a stress period with leakage to and from the aquifer (McDonald and Harbaugh, 1988). The Stream package accounts for leakage but allows flow to be routed through the river system only by a uniform, steady-state technique (Prudic, 1989). The coupling of BRANCH with MODFLOW expands the simulation capability to include one-dimensional routing of streamflow in a network of interconnected open channels while accounting for the effects of transient leakage between the aquifer and the stream. The MODBRANCH interface of MODFLOW and BRANCH is similar to the River and Stream packages of MODFLOW and processes the information passed from BRANCH through MODBRANCH using a method similar to that used in MODFLOW. The modified form of BRANCH, which is called through the MODBRANCH code, is known as BRANCH' (Branch prime).

The concept of initiating BRANCH' runs using MODFLOW is based on the need for coincident time periods for the coupled models. The time scale of variations in the surface-water flow is on the order of minutes and hours. Ground-water flow generally varies in hours, days, or months. Thus, it is necessary to allow multiple time intervals to pass in BRANCH' for each time step in MODFLOW. Each time MODFLOW runs one ground-water time step, BRANCH' is called from MODFLOW to simulate the

number of surface-water time intervals that correspond to the ground-water time step. This scheme requires that the surface-water time-interval size be less than or equal to the ground-water time step, and the ground-water time step must be an integral multiple of the surface-water time interval. The need to specify a surface-water time interval longer than the ground-water time step is considered almost non-existent, but frequently a surface-water time interval will be less than the ground-water interval. Determinations of relative time scales for surface and subsurface flow modeling based on the physical characteristics have been studied by Yen and Riggins (1991).

The computation of leakage between the stream and aquifer is included in MODFLOW; however, the current formulation of BRANCH does not include a leakage term. A scheme was developed where leakage was calculated separately in MODFLOW and BRANCH' as an implicit function of the stage in the river and the head in the corresponding aquifer. To conserve mass for coupled simulations, this scheme required a modification of the continuity equation originally used in BRANCH. Alternatively, when multiple BRANCH' time intervals occur within one MODFLOW time step, variations in river stage simulated by BRANCH' and occurring within the single MODFLOW time step could not be represented in the original formulation of MODFLOW, which only uses the values of river stage at the beginning and end of each MODFLOW time step. A floodwave could pass down the river channel in BRANCH' with BRANCH' simulations accounting for the ensuing riverbed leakage but without such leakage being accounted for by MODFLOW (fig. 1).

To account for leakage in the coupled models when BRANCH' time intervals do not equal MODFLOW time steps, a less numerically stable, but more accurate, scheme is used. Average leakage flow rates calculated by BRANCH' during a MODFLOW time step are computed and applied to MODFLOW simulations during the entire ground-water time step. The aquifer head at each BRANCH' time interval is linearly interpolated from the heads calculated by MODFLOW at the beginning and end of this time step. Although not as numerically stable at high leakage rates as the original implicit calculation of leakage in MODFLOW, this scheme maintains mass balance between the two models.

The scheme necessitates multiple iterations between the two models for each MODFLOW time step. Figure 2 shows how MODFLOW and BRANCH' interface and pass variables. Ground-water heads at the beginning and end of each new time step are initialized using heads computed at the end in the previous time step. BRANCH' is then called and, with the interpolated ground-water heads, the streamflow is calculated for the number of surface-water time intervals in the ground-water time step. The total leakage per BRANCH' time step is calculated simultaneously. After returning to MODFLOW, the single MODFLOW time step

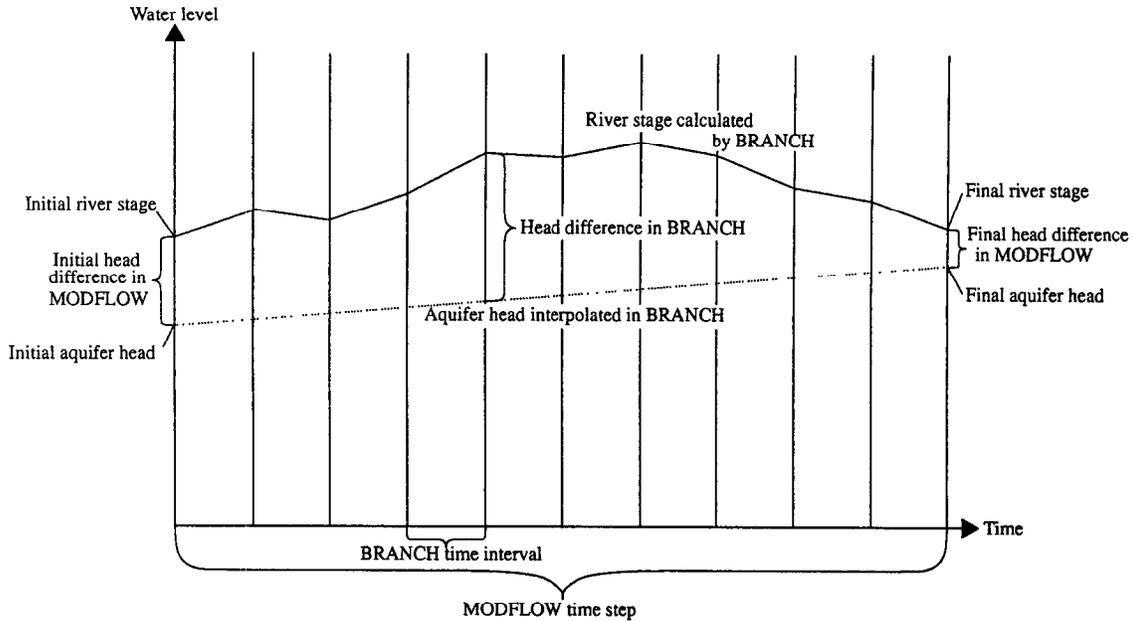


Figure 1. Computation of head differences in the BRANCH and MODFLOW models.

is simulated using leakage calculated by BRANCH', and a new estimate of ground-water heads is made. BRANCH' is called again, and stages and discharges in the channel are reset to their values at the beginning of the ground-water time step, and streamflow is recalculated with leakage based on the new estimate of ground-water heads at time-step end. This process is repeated until the difference in successive estimates of heads and stages drops below a user specified criteria. The model then advances to the next ground-water time step.

The locations in the aquifer corresponding to stream reaches are specified in the BRANCH' input. The head in each model aquifer cell is assumed to be the same throughout the entire cell. Each stream segment is assigned to an aquifer model cell; thus, no segment can span more than one cell, and a channel cross section is defined at each point where a river enters or leaves an aquifer model cell. Multiple river segments can occur within a cell, but inflow and outflow from each reach is considered to occur at the center of the cell. A typical arrangement of aquifer model cells and river segments is shown in figure 3. All leakage to and from a river segment is considered to occur only with the corresponding aquifer model cell.

To enhance the modular characteristics of the coupled model, BRANCH' was rearranged so that all of its array variables were allocated space in three main arrays in MODFLOW: a real and integer array, a character array, and a logical array. Thus, redimensioning of arrays is simpler and greatly reduces the number of common statements needed. (These statements transfer variables from routine to routine.) The sequence in which the modules in the MODBRANCH code are called from MODFLOW is shown in

figure 4. The amount of space and position each BRANCH' array uses in one of the main arrays is allocated in an allocation (AL) module. The original BRANCH code was split into a data entry module (RP) and a computational model (BRANCH'). BRANCH', which contains all of BRANCH except the data entry procedure, is called from a formulation module (FM) in MODFLOW, which adds the BRANCH' leakage to MODFLOW. The BRANCH' computational model is also called from a BUDGET module (BD), which calculates the cumulative flows to and from the stream and prints a summary at the end of the time interval.

The AL and RP modules are called at the beginning of the simulation. The FM module is called for every iteration between MODFLOW and BRANCH'; the FM module, in turn, calls BRANCH'. The BD module is called at the end of each ground-water time step along with BRANCH'.

MATHEMATICAL FORMULATION

The main modification made to the mathematical formulation in this coupled ground-water and surface-water model is the addition of the leakage terms to the original continuity equation in BRANCH. Several smaller changes were made to BRANCH in conjunction with the creation of the connection package MODBRANCH called from MODFLOW.

INCORPORATION OF LEAKAGE

The term for leakage or another inflow or outflow in MODFLOW had already been incorporated in the well,

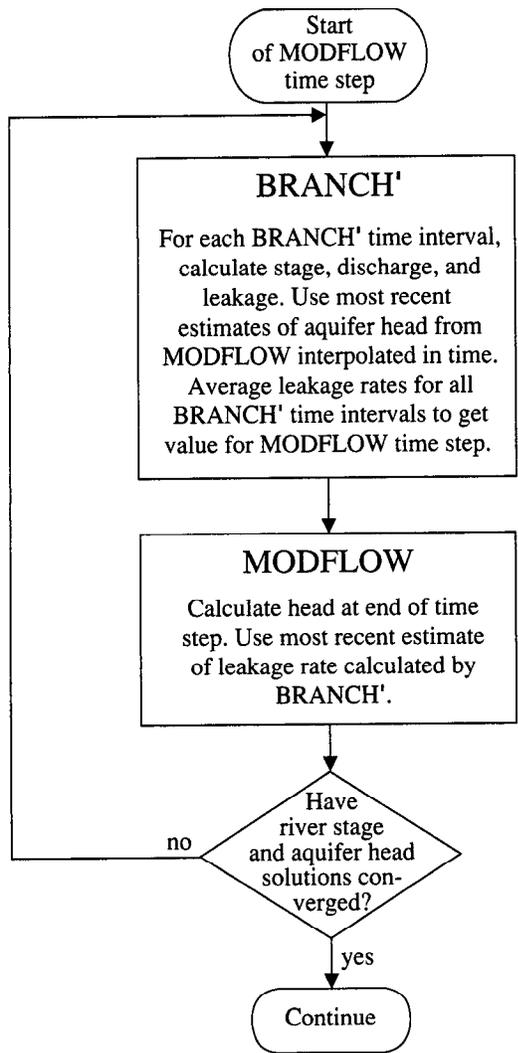


Figure 2. Iteration procedure between MODFLOW and BRANCH'.

river, stream, drain, and recharge packages. Leakage from BRANCH' is incorporated in the same fashion into MODFLOW; however, leakage has to be incorporated into the BRANCH' formulation. The original partial differential equation of continuity used in BRANCH is (Schaffranek and others, 1981)

$$B \frac{\partial Z}{\partial t} + \frac{\partial Q}{\partial L} = 0, \quad (1)$$

where B is channel topwidth, Z is stage in the channel, t is time, Q is flow rate in the channel, and L is longitudinal distance down the channel. When lateral inflows and outflows are included, the equation is (Schaffranek, 1987)

$$B \frac{\partial Z}{\partial t} + \frac{\partial Q}{\partial L} + q = 0, \quad (2)$$

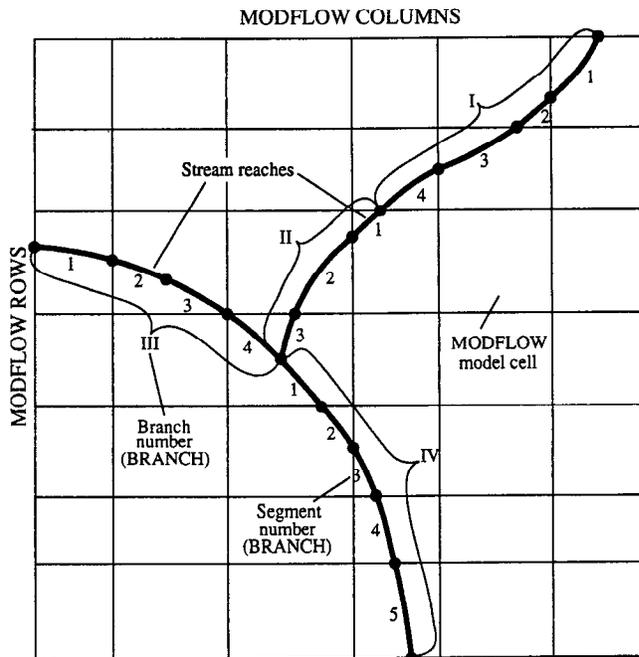


Figure 3. Arrangement of MODFLOW model cells and BRANCH stream reaches.

where q is outflow per unit length of channel. If outflow is the result of leakage to the aquifer and this leakage is considered to cross a riverbed with thickness b' and hydraulic conductivity K' , Darcy's law gives the leakage as

$$q = \frac{K'}{b'} B (Z - h), \quad (3a)$$

where h is head in the aquifer. This equation is used for leakage in the River and Stream packages. Equation 3a is equivalent to equation 63a in McDonald and Harbaugh (1988). The leakage perimeter of the channel is approximated by the topwidth. If the head in the aquifer is below the river bottom, the aquifer is partly saturated under the riverbed, and leakage is based on the head in the stream or

$$q = \frac{K'}{b'} B (Z - Z_{BOT}), \quad (3b)$$

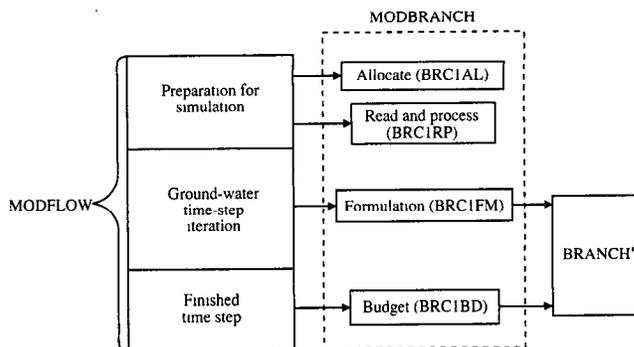


Figure 4. Schematic of module calling sequence.

where Z_{BOT} is elevation of river bottom. In the following equations, it will be assumed that if the aquifer head is below the river bottom, the value of h in the streamflow equation will be replaced by Z_{BOT} .

When equation 3a is included in equation 2, the resulting continuity equation can be put in finite-difference form with a similar format to that originally used in BRANCH (Schaffranek and others, 1981):

$$\begin{aligned} & \frac{\bar{B}}{B} \left[\frac{Z_{i+1}^{j+1} + Z_i^{j+1}}{2\Delta t} - \frac{Z_{i+1}^j + Z_i^j}{2\Delta t} \right] \\ & + \Theta \frac{Q_{i+1}^{j+1} - Q_i^{j+1}}{\Delta L_i} + (1 - \Theta) \frac{Q_{i+1}^j - Q_i^j}{\Delta L_i} \\ & + \frac{\chi}{2} \left[C_{i+1} B_{i+1}^{j+1} \left(Z_{i+1}^{j+1} - h^{j+1} \right) \right. \\ & \quad \left. + C_i B_i^{j+1} \left(Z_i^{j+1} - h^{j+1} \right) \right] \\ & + \frac{(1 - \chi)}{2} \left[C_{i+1} B_{i+1}^j \left(Z_{i+1}^j - h^j \right) \right. \\ & \quad \left. + C_i B_i^j \left(Z_i^j - h^j \right) \right] = 0, \end{aligned} \quad (4)$$

where C is K/b' , ΔL_i is length of channel segment from points i to $i+1$, Θ is weighting factor for spatial derivatives, χ is a weighting factor for averaged quantities, and \bar{B} is average channel topwidth from the previous time interval:

$$\bar{B} = \chi \frac{B_{i+1}^j + B_i^j}{2} + (1 - \chi) \frac{B_{i+1}^{j-1} + B_i^{j-1}}{2}. \quad (5)$$

The subscripts indicate location in space, such that i is the upstream node and $i+1$ is the downstream node. The superscripts indicate time of occurrence, such that j is the beginning of the time interval, $j+1$ is the end of the time interval, and $j-1$ is the beginning of the previous time interval.

Equation 4 is solved simultaneously for all nodes (points along the channel where cross sections are defined), with the finite-difference form of the momentum equation unchanged in BRANCH' (Schaffranek and others, 1981). Coefficients for a matrix solution were developed to use the same matrix solution that was already implemented in BRANCH, putting equation 4 in the form

$$Q_{i+1}^{j+1} + \gamma Z_{i+1}^{j+1} - Q_i^{j+1} + \alpha Z_i^{j+1} = \delta, \quad (6a)$$

$$\text{where } \gamma = \frac{\bar{B}\Delta L_i}{2\Delta t\Theta} + \frac{\chi C_{i+1} B_{i+1}^{j+1} \Delta L_i}{2\Theta}, \quad (6b)$$

$$\alpha = \frac{\bar{B}\Delta L_i}{2\Delta t\Theta} + \frac{\chi C_i B_i^{j+1} \Delta L_i}{2\Theta}, \quad (6c)$$

$$\begin{aligned} \text{and } \delta = & -\frac{1-\Theta}{\Theta} \left(Q_{i+1}^j - Q_i^j \right) + \left[\frac{\bar{B}\Delta L_i}{2\Delta t\Theta} - (1-\chi) \right. \\ & \left. \left(C_{i+1} B_{i+1}^j \right) \frac{\Delta L_i}{2\Theta} \right] Z_{i+1}^j \\ & + \left[\frac{\bar{B}\Delta L_i}{2\Delta t\Theta} - (1-\chi) \left(C_i B_i^j \right) \frac{\Delta L_i}{2\Theta} \right] Z_i^j + \frac{\Delta L_i}{2\Theta} \\ & \left[\chi \left(C_{i+1} B_{i+1}^{j+1} h^{j+1} + C_i B_i^{j+1} h^{j+1} \right) \right. \\ & \left. + (1-\chi) \left(C_{i+1} B_{i+1}^j h^j + C_i B_i^j h^j \right) \right] \end{aligned} \quad (6d)$$

In the scheme without leakage, γ and α are the same quantity (Schaffranek and others, 1981). However, the coefficients γ , α , and δ can be placed in the same positions in the matrix as their nonleakage predecessors. This provides a similar form of the matrix of the flow equations in the i th segment:

$$\begin{bmatrix} 1 & \zeta \\ \gamma & 1 \end{bmatrix} \begin{bmatrix} Z_{i+1}^{j+1} \\ Q_{i+1}^{j+1} \end{bmatrix} - \begin{bmatrix} 1-\omega \\ -\alpha & 1 \end{bmatrix} \begin{bmatrix} Z_i^{j+1} \\ Q_i^{j+1} \end{bmatrix} = \begin{bmatrix} \varepsilon \\ \delta \end{bmatrix}, \quad (7)$$

where ζ , ω , and ε are coefficients in the momentum equation. The preexisting method of saving computational effort in BRANCH by branch transformation as described in Schaffranek and others (1981) is maintained in BRANCH' using the coefficients in equation 7.

The BRANCH' model is modified to incorporate channel-bed leakage to and from the aquifer. The only variable in the computation scheme upon which leakage depends is the stage Z . The only input needed from the ground-water model is the aquifer heads h , which are fixed values for the solution of equation 7. The feedback of leakage quantity occurring in BRANCH' is returned to MODFLOW so it can calculate new values of h . The leakage quantities for all the BRANCH' time intervals during one MODFLOW time step must be calculated and averaged in BRANCH'. This averaging process is accomplished using

$$\frac{\overline{q\Delta L}_i}{NTSAQ} = \frac{\Delta L_i}{NTSAQ} \sum_{j=T_s}^{T_s+NTSAQ} \frac{\chi}{2} \left[C_{i+j} B_{i+j}^{j+1} \left(Z_{i+j}^{j+1} - h^{j+1} \right) + C_{i+j} B_{i+j}^j \left(Z_{i+j}^j - h^j \right) \right] + \frac{1-\chi}{2} \left[C_{i+1} B_{i+1}^j \left(Z_{i+1}^j - h^j \right) + C_i B_i^j \left(Z_i^j - h^j \right) \right], \quad (8)$$

where T_s is the starting time interval when BRANCH' is entered from MODFLOW, and $NTSAQ$ is the number of BRANCH' time intervals in one MODFLOW time step. For example, if the MODFLOW time step is 1 hour, and the BRANCH' time interval is 5 minutes, then $NTSAQ$ is 12.

The quantity derived in equation 8 is the average leakage flow rate into or out of reach i of the stream during the MODFLOW time step ($NTSAQ$ BRANCH' times intervals). It can be transferred directly back to MODFLOW and added to the flow in the aquifer model cell. The three-dimensional ground-water flow equation takes the form (McDonald and Harbaugh, 1988)

$$\frac{\partial}{\partial x} \left(K_x \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y} \left(K_y \frac{\partial h}{\partial y} \right) + \frac{\partial}{\partial z} \left(K_z \frac{\partial h}{\partial z} \right) - W = S_s \frac{\partial h}{\partial t}, \quad (9)$$

where x , y , and z are coordinate directions, K_x , K_y , and K_z are hydraulic conductivities of the aquifer in these coordinate directions, S_s is specific storativity, and W is volumetric flux per unit volume.

The W term corresponds to a leakage quantity or other inflow or outflow. In a report by McDonald and Harbaugh (1988), the derivation of the finite-difference form of equation 9 that is used in MODFLOW is shown. This derivation yields the equation

$$\begin{aligned} & V_{i,j,k-1/2} h_{i,j,k-1}^m + U_{i-1/2,j,k} h_{i-1,j,k}^m + R_{i,j-1/2,k} h_{i,j-1,k}^m \\ & + (-V_{i,j,k-1/2} - U_{i-1/2,j,k} - R_{i,j-1/2,k} - R_{i,j+1/2,k} - U_{i+1/2,j,k} - V_{i,j,k+1/2} + H_{COEF_{i,j,k}}) h_{i,j,k}^m \\ & + R_{i,j+1/2,k} h_{i,j+1,k}^m + U_{i+1/2,j,k} h_{i+1,j,k}^m + V_{i,i,k+1/2} h_{i,i,k+1}^m \\ & h_{i,j,k+1}^m = RHS_{i,j,k}, \end{aligned} \quad (10)$$

where i , j , and k are row, column, and layer indices, m is time level,

$$V_{i,j,k} = \frac{K_{z_{i,j,k}} \Delta X_i \Delta Y_j}{\Delta Z_k},$$

$$U_{i,j,k} = \frac{K_{x_{i,j,k}} \Delta Y_j \Delta Z_k}{\Delta X_i},$$

$$R_{i,j,k} = \frac{K_{y_{i,j,k}} \Delta X_i \Delta Z_k}{\Delta Y_j},$$

$$H_{COEF_{i,j,k}} = P_{ijk} - \frac{S_{s_{i,j,k}} \Delta X_i \Delta Y_j \Delta Z_k}{t^m - t^{m-1}},$$

$$RHS_{i,j,k} = -F_{ijk} - \frac{S_{s_{i,j,k}} h_{i,j,k}^{m-1} \Delta X_i \Delta Y_j \Delta Z_k}{t^m - t^{m-1}},$$

P_{ijk} is a head-dependent inflow term, and F_{ijk} is the inflow term.

The term F_{ijk} is the flow rate ($L^3 t^{-1}$) from an external source into the aquifer model cell i, j, k . Thus, the $\overline{q\Delta L}_i$ term calculated in BRANCH' by equation 8 for a specific river segment can be passed to MODFLOW and added to the F_{ijk} term in equation 10 for the aquifer model cell containing the river segment.

If there is only one BRANCH' time interval in the MODFLOW time step (same time-scale lengths), leakage can be calculated implicitly in MODFLOW instead of passing $\overline{q\Delta L}_i$ from BRANCH'. This scheme is more stable numerically because it adds terms to the diagonal of the MODFLOW matrix, making it more diagonally dominant. This is achieved by setting the terms in equation 10 as follows:

$$P_{ijk} = - \left(C_{i+1} B_{i+1}^{j+1} + C_i B_i^{j+1} \right) \frac{\Delta L_i}{2}, \quad (11)$$

$$F_{ijk} = \left(C_{i+1} B_{i+1}^{j+1} Z_{i+1}^{j+1} + C_i B_i^{j+1} Z_i^{j+1} \right) \frac{\Delta L_i}{2}. \quad (12)$$

These terms are fully forward weighted in time, as is the rest of the MODFLOW formulation. Thus, to be consistent, the BRANCH' formulation should have a forward-weighted leakage term ($\chi=1.0$) in this case.

These equations are used in the module calling sequence shown in figure 4. Values of h are passed from MODFLOW to BRANCH' for solving the channel flow equation 4 along with the momentum equation for values of Z and Q . After this solution is made iteratively, the leakage rate equation 8 is used to determine $\overline{q\Delta L}_i$ for all river segments for the number of BRANCH' time intervals that occur during the MODFLOW time step. If multiple BRANCH' time intervals occur in one MODFLOW time step, these $\overline{q\Delta L}_i$ values are passed back to MODFLOW and used as the F_{ijk} inflow value in the ground-water flow equation 10 for the aquifer model cell containing the river segment. Alternatively, if MODFLOW and BRANCH' have the same time-step and time-interval lengths, components P_{ijk} and F_{ijk} are calculated by equations 11 and 12 and transferred to the ground-water flow equation 10. Solving equation 10 provides revised values of h to be passed back to BRANCH' and the process is repeated. This process is continued until the values of h and Z show no significant change from iteration to iteration, thus, signaling the com-

pletion of a MODFLOW time step. Four to nine iterations are usually sufficient for convergence of MODFLOW and BRANCH'. More iterations are usually necessary when either the ground-water or surface-water system changes rapidly.

DRYING AND REWETTING OF RIVER CHANNELS

Another option that is included in the coupled model is the representation of the drying and rewetting of the river channel. This option allows the modeling of an intermittent flow system where the river is periodically fed by ground water or is completely drained by leakage to the aquifer. Numerical stability is the most common problem that occurs when trying to simulate a dry condition with the unsteady flow equation. The continuity equation (eq. 4) creates no numerical problems at small values of B , but the momentum equation may be unstable under certain conditions. The finite-difference form of the momentum equation is (Schaffranek and others, 1981)

$$\begin{aligned} & \frac{1}{g\bar{A}} \left[\frac{Q_{i+1}^{j+1} + Q_i^{j+1}}{2\Delta t} - \frac{Q_{i+1}^j + Q_i^j}{2\Delta t} \right] + \frac{2\beta\bar{Q}}{g\bar{A}^{-2}} \left[\Theta \frac{Q_{i+1}^{j+1} - Q_i^{j+1}}{\Delta L_i} \right. \\ & + (1 - \Theta) \frac{Q_{i+1}^j - Q_i^j}{\Delta L_i} \left. \right] - \frac{\beta\bar{Q}^{-2}}{g\bar{A}^{-3}} \frac{\bar{A}_{i+1}^{j+1} - \bar{A}_i^{j+1}}{\Delta L_i} + \Theta \frac{Z_{i+1}^{j+1} - Z_i^{j+1}}{\Delta L_i} \\ & + (1 - \Theta) \frac{Z_{i+1}^j - Z_i^j}{\Delta L_i} + \frac{k|\bar{Q}|}{A r^{-2.4/3}} \left[\chi \frac{Q_{i+1}^{j+1} + Q_i^{j+1}}{2} + (1 - \chi) \right. \\ & \left. \frac{Q_{i+1}^j + Q_i^j}{2} \right] - \frac{\xi\bar{B}}{g\bar{A}} U_a^2 \cos \phi = 0, \end{aligned} \quad (13)$$

where the overbar indicates quantities averaged from the previous time interval; A is the cross-sectional area of channel; β is the momentum coefficient; g is gravitational acceleration; k is the friction factor $(n/1.49)^2$ based on Manning's equation, in inch-pound units or n^2 in metric units; r is the hydraulic radius; ξ is $C_d \rho_a / \rho_w$ (ρ_a is air density and ρ_w is water density); C_d is the water surface drag coefficient; U_a is the wind speed; and ϕ is the angle between wind direction and channel orientation.

Equation 13 has cross-sectional area, \bar{A} , in the denominator of many terms and would be unstable for small values of \bar{A} . To compensate for this problem, a scheme was developed that retains a small flow in the channel, increases the frictional resistance of the streambed to allow as little discharge as possible, and eliminates any leakage to the aquifer. Flow continuity is retained in the channel using this

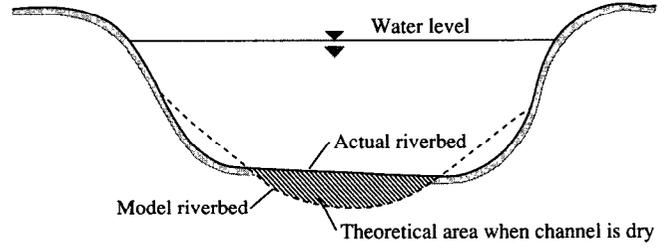


Figure 5. Theoretical channel configuration to allow for channel drying.

scheme. In addition, there is no rapid change in the mathematical formulation for the dry channel that would induce instabilities, and a rewetting of the channel can be initiated by a rise in stream stage. The creation of a small theoretical area below the riverbed to provide data for equation 13 is a simple but necessary addition to the cross-sectional geometry table in BRANCH'. A diagram of this channel configuration is shown in figure 5. When the stage in the channel drops into this theoretical area, the frictional resistance in the momentum equation is increased by a user-defined amount.

The increase to streambed resistance is accomplished using the coefficients λ , μ , σ for equation 13, such that

$$\lambda = \frac{\Delta L_i}{2\Delta t \Theta g\bar{A}}, \quad (14a)$$

$$\mu = \frac{2\beta\bar{Q}}{g\bar{A}}, \quad (14b)$$

$$\sigma = \frac{\chi \Delta L_i k |\bar{Q}|}{2\Theta A r^{-2.4/3}}, \text{ and} \quad (14c)$$

$$\begin{aligned} \varepsilon = & \left(\lambda - \sigma \frac{(1-\chi)}{\chi} \right) (Q_{i+1} + Q_i) + \mu \frac{(1-\Theta)}{\Theta} [Q_{i+1} - Q_i] \\ & - \frac{1-\Theta}{\Theta} [Z_{i+1} - Z_i] + \frac{\beta\bar{Q}^{-2}}{\Theta g\bar{A}} \\ & [A_{i+1}^{j+1} - A_i^{j+1}] + \frac{\xi \Delta L_i \bar{B}}{\Theta g\bar{A}} U_a^2 \cos \alpha. \end{aligned} \quad (14d)$$

With $\zeta = \lambda + \sigma + \mu$ and $\omega = \lambda + \sigma - \mu$, equation 13 becomes,

$$\zeta Q_{i+1}^{j+1} + Z_{i+1}^{j+1} + \omega Q_i^{j+1} - Z_i^{j+1} = \varepsilon. \quad (15)$$

These are the ζ , ω , and ε coefficients used in equation 7.

The coefficient σ contains the friction term. When a river stage drops below the elevation of the river bottom, σ is multiplied by a dry channel friction multiplier (DCFM). The value of DCFM should be as large as possible without

creating computational instabilities. An increase of the DCFM causes a decrease in flow when the channel is dry.

If σ is multiplied by DCFM when the stage drops below the riverbed, two problems can occur: (1) the computational instabilities may cause an oscillation between two alternate solutions without convergence—where the stage is below the riverbed and water is flowing at high friction and where the stage is above the riverbed and water is flowing at normal friction; and (2) a jump in stage results when the channel rewets because of the sudden drop in friction.

These problems are solved by varying DCFM gradually as the channel dries and wets. When progressively higher values were applied to DCFM with decreasing stage below the riverbed, greater oscillations occurred in the BRANCH' solution as the stage fluctuated above and below the riverbed. The most stable solution seems to be gradating DCFM by time. If the channel is currently dry and was dry in the last time interval, DCFM is doubled with an upper limit at the DCFM value selected by the user. If the channel is rewet, DCFM is halved each time interval. BRANCH' performs the gradation from first drying to full value of DCFM in five time intervals and from full value of DCFM back to the lowest value also in five intervals. When the appropriate maximum DCFM value is chosen by the user, smooth transitions occur between dry and wet conditions.

Several concerns are raised by the empirical nature of this solution. Because the gradations in DCFM are performed in successive time intervals, time-interval length will affect the rate of drying and rewetting frictional changes. This is not a desirable condition, but trial runs (see example problem 3) indicated that this effect is concentrated around the times of transition between wet and dry and dry and wet and does not propagate into the rest of the solution. Although at full friction negligible flows should be expected, in practice DCFM is so low that significant flows may occur just before rewetting. This condition should probably be considered as wet by the program user.

The BRANCH' user should adjust the selected DCFM value to calibrate the drying and rewetting curves. A value of 100 worked well in the sample runs, but DCFM may vary according to the variability of streamflow and channel conditions. When interpreting results, recognize that the drying and rewetting process is empirically calibrated, and the exact moment the channel runs dry cannot be determined from model results. However, exact times of channel drying and rewetting also are difficult to determine in the field.

STEADY-STATE SIMULATION

Because BRANCH uses the full, unsteady flow St. Venant equation, it may seem counterproductive to make modifications in order to simulate steady-state riverflow. However, it is often desirable to model the steady-state aquifer condition in MODFLOW. Accordingly, a steady-

state option was developed for BRANCH' that will make it compatible with MODFLOW. This is achieved by removing the time-dependent terms in the continuity and momentum equations 4 and 13, which is equivalent to setting Δt in the denominators to infinity. This is done in the coefficients by subtracting

$$\frac{\bar{B}\Delta L_i}{2\Delta t\Theta}$$

from γ and α (eqs. 6b and 6c), and by subtracting

$$\frac{\bar{B}\Delta L_i}{2\Delta t\Theta} (Z'_{i+1} + Z'_i)$$

from δ (eq. 6d), and setting λ to zero (eq. 14a). The equations are solved by the previously described method, and the leakage effects are retained.

MODEL DOCUMENTATION

The method used in integrating the BRANCH' and MODFLOW models was to create an interface code (MODBRANCH) called by MODFLOW that passes information between MODFLOW and BRANCH' and calls BRANCH'. The MODBRANCH code consists of four modules called by MODFLOW. The last two modules each call the model BRANCH'. Subroutines called by BRANCH' are not described in detail in this report. Descriptions of these subroutines can be found in the report by Schaffranek and others (1981).

MODBRANCH modules:

- BRC1AL allocates space for data arrays used by BRANCH'.
- BRC1RP reads input data for BRANCH'. Consists of the data reading section of the original BRANCH.
- BRC1FM calls BRANCH' and adds leakage to the RHS terms in each aquifer model cell containing a river reach.
- BRC1BD calls BRANCH' to calculate rates and accumulated volumes over a MODFLOW time step.

Model called by MODBRANCH modules:

- BRANCH' called by BRC1FM and BRC1BD modules. Simulates the unsteady flow in networks of reaches composed of interconnecting channels with leakage to the aquifer. This code consists of the computational section of the open-channel flow model BRANCH with modifications for leakage, drying and rewetting of reaches, and the steady-state option.

MODIFICATIONS TO MAIN CODE

Several modifications to the MODFLOW main program are necessary to allow it to use the MODBRANCH code. These modifications are as follows:

1. Introduce Y, YC, and YL arrays. This is done by inserting the following commands at the program beginning:
COMMON Y (NUMERICAL LENGTH OF Y VECTOR)
DOUBLE PRECISION Y
DOUBLE PRECISION YUMMY
EQUIVALENCE (YUMMY, Y(1))
CHARACTER*4 YC (NUMERICAL LENGTH OF YC VECTOR)
LOGICAL*4 YL (NUMERICAL LENGTH OF YL VECTOR)
2. Introduce common statements. There are 11 common cells and some dimension statements pertaining to the variables in the MODBRANCH code. These were placed at the beginning of the MODFLOW main program so appropriate variables can be passed between routines. This is given in the listing of the modified main program (table 1).
3. Define lengths of vectors. Between comments C1 and C2 in the main code, insert the vector lengths:
LENY = (NUMERICAL LENGTH OF Y VECTOR)
LENYC = (NUMERICAL LENGTH OF YC VECTOR)
LENYL = (NUMERICAL LENGTH OF YL VECTOR)
4. Introduce calls to MODBRANCH modules. The four modules in the MODBRANCH code are called at the appropriate location.

Between comments C4 and C5 is added:

```

IF(IUNIT(?).GT.0)
1 CALL BRC1AL (LENY,LENYC,LENYL,LINDAT,
LIZDAT,LIQDAT,LITQMA,LITQMI,
1 LQMAX,LQMIN,LQSUM,LZQMIN,LZQMAX,
LAQMAX,LAQMIN,LA,LZ,LQ,LZP,LQP,
2 LAP,LBP,LRP,LB,LR,LBT,LBTP,LXSTAT,LDX,
LT,LRN,LWANGL,LGDATU,LORIEN,
3 LBETVE,LSUMET,LSUMCZ,LSC-
ZQS,LSZQET,LITYPO,LZA,LAA,LBB,LBS,LIPT,
4 LQA,LTA,LETA,LFUNET,LROW,LAM,LBMX,
LBRNAM,LIJF,LIJT,LNSEC,LXSKT,
5 LPLTBC,LPRTXS,LPRTBC,LPRTSU,LPPLTB,
LITYPE,LIBJNC,LNDATA,LIZQB V,
6 LISTAT,LKTTDB,LZQ,LDTT,LDATUM,
LZQBVC,LZQPMI,LLARBP,LZHIGH,
7 LZLOW,LLINPR,LARBER,LCLK,LZBOT,
MXBH,NSEC,MAXS,MXTDBC,
8 MAXZBD,MAXCZQ,MXWIND,MXJN,NSEC,
MXPT,MXMD,MAXQBD,MAXMZQ,ITMUNI,
9 LWINDS,LWINDD,LIDX,LICT,LW,LU,LUU,
LBU,LBUU,LZSAV,LQSAV,LZPSAV,

```

- 1 LQPSAV,IUNIT(?),IOUT,TFCTR,LISTRM, LZPL,LSLKG,IBEGIN,LZN,
- 2 LQLSUM,NELAP,LITRIA,DCFM)

Between comments C6 and C7 is added:

```

IF(IUNIT(?).GT.0) CALL BRC1RP
& (Y(LINDAT),Y(LIZDAT),Y(LIQDAT),Y(LITQ-
MA),Y(LITQMI),Y(LQMAX),
1 Y(LQMIN),Y(LQSUM),Y(LZQMIN),Y(LZQ-
MAX),(LAQMAX),Y(LAQMIN),Y(LA),
2 Y(LZ),Y(LQ),Y(LZP),Y(LQP),Y(LAP),Y(LBP),
Y(LRP),Y(LB),Y(LR),
3 Y(LBT),Y(LBTP),Y(LXSTAT),Y(LDX),Y(LT),
Y(LRN),Y(LWANGL),Y(LGDATU),
4 (LORIEN),Y(LBETVE),Y(LSUMET),Y(LS-
UMCZ),Y(LSCZQS),Y(LSZQET),
5 YC(LITYPO),Y(LZA),Y(LAA),Y(LBB),Y(LBS),
Y(LIPT),Y(LQA),Y(LTA),
6 Y(LETA),Y(LFUNET),YC(LBRNAM),Y(LIJF),
7 Y(LIJT),Y(LNSEC),Y(LXSKT),Y(LPLTBC),
Y(LPRTXS),Y(LPRTBC),Y(LPRTSU),
8 Y(LPPLTB),YC(LITYPE),Y(LIBJNC),Y(LN-
DATA),Y(LIZQB V),Y(LISTAT),
9 Y(LKTTDB),Y(LZQ),Y(LDTT),Y(LDATUM),
Y(LZQBVC),Y(LZQPMI),
1 YL(LLARBP),YL(LZHIGH),YL(LZLOW),YL-
(LLINPR),YL(LARBER),Y(LCLK),
2 Y(LZBOT),MXBH,MXJN,MAXS,MXPT,MXT-
DBC,MXMD,MAXZBD,
3 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,
MAXBD,MXOTDT,Y(LWINDS),Y(LWINDD),
4 Y(LICT),Y(LW),IUNIT(?),IOUT,Y(LISTRM))

```

Between comments C7C2A and C7C2B and after the call to BAS1FM is added:

```

IF(IUNIT(?).GT.0) CALL BRC1FM
& (Y(LINDAT),Y(LIZDAT),Y(LIQDAT),Y(LITQ-
MA),Y(LITQMI),Y(LQMAX),
1 Y(LQMIN),Y(LQSUM),Y(LZQMIN),Y(LZQ-
MAX),Y(LAQMAX),Y(LAQMIN),Y(LA),
2 Y(LZ),Y(LQ),Y(LZP),Y(LQP),Y(LAP),Y(LBP),
Y(LRP),Y(LB),Y(LR),
3 Y(LBT),Y(LBTP),Y(LXSTAT),Y(LDX),Y(LT),
Y(LRN),Y(LWANGL),Y(LGDATU),
4 Y(LORIEN),Y(LBETVE),Y(LSUMET),Y(LS-
UMCZ),Y(LSCZQS),Y(LSZQET),
5 YC(LITYPO),Y(LZA),Y(LAA),Y(LBB),Y(LBS),
Y(LIPT),Y(LQA),Y(LTA),
6 Y(LETA),Y(LFUNET),Y(LROW),Y(LAM),
Y(LBMX),YC(LBRNAM),Y(LIJF),
7 Y(LIJT),Y(LNSEC),Y(LXSKT),Y(LPLTBC),
Y(LPRTXS),Y(LPRTBC),Y(LPRTSU),
8 Y(LPPLTB),YC(LITYPE),Y(LIBJNC),Y
(LNDATA),Y(LIZQB V),Y(LISTAT),

```

Table 1. Example listing of a modified main program

```

C *****
C MAIN CODE FOR MODULAR MODEL -- 4/1/91
C BY MICHAEL G. MCDONALD AND ARLEN W. HARBAUGH
C-----VERSION 1700 1APR1991 MAIN1 MODIFIED TO INTERFACE WITH THE
C UNSTEADY OPEN CHANNEL FLOW MODEL BRANCH
C *****
C
C SPECIFICATIONS:
C -----
C COMMON X(30000)
C COMMON Y(50000)
C DOUBLE PRECISION Y
C COMMON /FLWCOM/LAYCON(80)
C CHARACTER*4 HEADNG, VBNM
C DIMENSION HEADNG(32), VBNM(4,20), VBVL(4,20), IUNIT(24)
C DOUBLE PRECISION DUMMY
C EQUIVALENCE (DUMMY,X(1))
C DOUBLE PRECISION YUMMY
C EQUIVALENCE (YUMMY,Y(1))
C CHARACTER*4 YC(1000)
C LOGICAL*4 YL(1000)
C BEGIN COMMON_COMCON =====
C
C CHARACTER*2 IUNET, OUNIT
C INTEGER*4 NBCH, NJNC, NBND, NSTEPS, IRDGEO, NIT, IPROPT, IPLOPT, IPLDEV,
1 IPRMSG, IPLMSG, IEXOFT, INHR, INMN, IDTM, IWRTIC, IRDIC, NUMCOM, INWIND,
2 TYPETA, OTTDDDB, ISMOPT, NTDIOF, IRDNXT, IARDEM
C REAL THETA, QQTOL, ZZTOL, WSPEED, WDIREC, WSDRAG, H2ODEN, CHI, QZCONV,
1 ZDATUM, DT, G, AIRDEN, GLETA, GLBETA, ETAMIN, ETAMAX, TOLERR
C COMMON /COMCON/ NBCH, NJNC, NBND, NSTEPS, IRDGEO, NIT, IPROPT,
1 IPLOPT, IPLDEV, IPRMSG, IPLMSG, IEXOFT, INHR, INMN, IDTM, IWRTIC,
2 IRDIC, NUMCOM, INWIND, THETA, QQTOL, ZZTOL, WSPEED, WDIREC, WSDRAG,
3 H2ODEN, CHI, ZDATUM, IUNET, OUNIT, TYPETA, OTTDDDB, ISMOPT, G, QZCONV, DT,
4 AIRDEN, IARDEM, NTDIOF, IRDNXT, GLETA, GLBETA, ETAMIN, ETAMAX, TOLERR
C
C END COMMON_COMCON =====
C BEGIN COMMON_DTYPES =====
C
C CHARACTER*2 DTYPE, ZTYPE, QTYPE, ATYPE, BTYPE, ZPTYPE, QPTYPE, DPTYPE
C COMMON /DTYPES/ DTYPE, ZTYPE, QTYPE, ATYPE, BTYPE, ZPTYPE, QPTYPE, DPTYPE
C
C END COMMON_DTYPES =====
C BEGIN COMMON_UNITS =====
C
C CHARACTER*2 IBLK, UNIT, EN, ME, MT, FT, TUNIT, DC
C COMMON /UNITS/ IBLK, UNIT, EN, ME, MT, FT, TUNIT, DC
C
C END COMMON_UNITS =====
C BEGIN COMMON_LUNUMS =====
C
C INTEGER*4 READER, PRINTR, PUNCH, DSREF, TDDATA, LUPTRK, LUIFLO, LUIVOL,
1 LUGEOM, LUINIT, LUCVOL
C COMMON /LUNUMS/ READER, PRINTR, PUNCH, DSREF, TDDATA, LUPTRK, LUIFLO,
1 LUIVOL, LUGEOM, LUINIT, LUCVOL
C
C END COMMON_LUNUMS =====
C BEGIN COMMON_DADCOM =====
C
C INTEGER*2 LISTB, LISTA, STRIP, RTCODE
C COMMON /DADCOM/ LISTB, LISTA, STRIP, RTCODE

```

Table 1. Example listing of a modified main program—Continued

```

C
C END COMMON_DADCOM =====
C BEGIN COMMON_DAYPMO =====
C
      INTEGER*2 DPERM(12)
      COMMON /DAYPMO/ DPERM
C
C END COMMON_DAYPMO =====
C BEGIN COMMON_LOGICS =====
C
      LOGICAL*4 PRTMSG, NOCONV, ERROR, OPLOTS, FOUND, NOEXTP,
      1 NOPRIT, DAYSUM, MOREBD, DTPRT, PTPLT, DAOPEN, STAGES, MODETA
      COMMON /LOGICS/ PRTMSG, NOCONV, ERROR, OPLOTS, FOUND, NOEXTP,
      1 NOPRIT, DAYSUM, MOREBD, DTPRT, PTPLT, DAOPEN, STAGES, MODETA
C
C END COMMON_LOGICS =====
C BEGIN COMMON_BCTIME =====
C
      INTEGER*4 IETIME, NETIME
      INTEGER *2 IRDPDY, IYR, IMO, IDA, IHR, IMN, NYR, NMO, NDA, NHR, NMN
      COMMON /BCTIME/ IETIME, NETIME, IRDPDY, IYR, IMO, IDA, IHR, IMN,
      1 NYR, NMO, NDA, NHR, NMN
C
C END COMMON_BCTIME =====
C BEGIN COMMON_DATIME =====
C
      INTEGER*4 KYR, KMO, KDA, KHR, KMN, M, KYRS, KMOS, KDAS, KHRS, KMNS
      COMMON /DATIME/ KYR, KMO, KDA, KHR, KMN, M, KYRS, KMOS, KDAS, KHRS, KMNS
C
C END COMMON_DATIME =====
C BEGIN COMMON_NETWORK =====
C
      CHARACTER*80 NETNAM
      COMMON /NETWORK/ NETNAM
C
C END COMMON_NETWORK =====
C BEGIN COMMON_MODBRCH =====
C
      COMMON /MODBRCH/ TWOC SQ, IDTPDY, TWOG, CW, II, ONECHI, DCHI, DTHETA,
      1 IBCH, IJZPBC, IJQPBC, DCFM1, KKITER
C
C END COMMON_MODBRCH =====
      INTEGER*4 MXBH, MXJN, MAXS, MXPT, MXTDBC, MXMD, MAXZBD,
      1 MAXQBD, MAXCZQ, MAXMZQ, MXWIND, MAXBD, MXOTDT, KTTDBC
      LOGICAL*4 ARBERR
      REAL*8 C1, C2, C3, C4, UUIJP1, UUIJP2, UUIJP3, UUIJP4
      REAL LAMBDA, MU, SETA, WDTT, TWOC SQ, TWOG, CW, ONECHI, DCHI, DTHETA, TH, WIND
      INTEGER*4 IAR, I, J, K, L, II, IJ, NS, KT, IS, N, NWREAD, NWDATA, INTDBC, IDTPDY
      REAL QTOL, ZTMIN, ZTMAX, ZPMIN, QPMIN, DXMIN, DXMAX
      COMMON /LIMITS/ QTOL, ZTMIN, ZTMAX, ZPMIN, QPMIN, DXMIN, DXMAX
      INTEGER*2 JYR, JMO, JDA, JHR, JMN, MYR, MMO, MDA, MHR, MMN
      INTEGER*4 ND, NDFIRT, NDPART, JETIME, N TSAQ
      COMMON /PARTIM/ ND, NDFIRT, NDPART, JETIME, JYR, JMO, JDA, JHR, JMN,
      1 MYR, MMO, MDA, MHR, MMN
      CHARACTER*80 COMENT(9)
      COMMON /CMMNT/ COMENT
      CHARACTER*2 IDETA(7)
      COMMON /ETASYM/ IDETA
      REAL WRATIO, DTZERO, ZTEMP, QTEMP, ZIJ, QIJ, DXIJ, QIJP1, ZIJP1, APZPIJ,
      1 BPZPIJ, BTZPIJ, RPZPIJ, BAVG, BTAVG, AAVG, RAVG, QAVG, ZAVG, BETCOR,

```

Table 1. Example listing of a modified main program—Continued

```

2      RNIJ, AAVGSQ, AAVGCU, SIGMA, EPSLON, ZETA, OMEGA, GAMMA, DELTA, DET,
3      DZDT, DQDXC, DQDT, DQDXM, DADX, DZDX, FRIC, ZQPIJ, BIGQ, BIGZ, ZTOL, SOLPDT
4      , ALPHA
      INTEGER*4 IBCH, IJZPBC, IJQPBC, KTMATS, LASTN, IJP1, NSM1, JP1, IJ2, IJ4,
1      IJ4P1, IJ4P2, IJ4P3, IJ4P4, IJ2P1, IJ2P2, I2, I4, I4P1, I4P2, I4P3, I4P4,
2      I2P1, I2P2, NN, MM, NNN, NBPJ, M0, IBIGZ, JBIGZ, IBIGQ, JBIGQ, IJPNS, ICHK
C      -----
C
C1-----SET SIZE OF X, Y, YC, YL ARRAYS. REMEMBER TO REDIMENSION ARRAYS.
      LENX=30000
      LENY=50000
      LENYC=1000
      LENYL=1000
C
C2-----ASSIGN BASIC INPUT UNIT AND PRINTER UNIT.
      INBAS=15
      IOUT=16
C
C3-----DEFINE PROBLEM ROWS, COLUMNS, LAYERS, STRESS PERIODS, PACKAGES
      CALL BAS1DF (ISUM, HEADNG, NPER, ITMUNI, TOTIM, NCOL, NROW, NLAY,
1      NODES, INBAS, IOUT, IUNIT)
C
C4-----ALLOCATE SPACE IN "X, Y, YL, YC," ARRAYS.
      CALL BAS1AL (ISUM, LENX, LCHNEW, LCHOLD, LCIBOU, LCCR, LCCC, LCCV,
1      LCHCOF, LCRHS, LCDELRL, LCDELCL, LCSTRT, LCBUFF, LCIOFL,
2      INBAS, ISTRT, NCOL, NROW, NLAY, IOUT)
      IF (IUNIT(1) .GT. 0) CALL BCF1AL (ISUM, LENX, LCSC1, LCHY,
1      LCBOT, LCTOP, LCSC2, LCTRPY, IUNIT(1), ISS,
2      NCOL, NROW, NLAY, IOUT, IBCFCB)
      IF (IUNIT(2) .GT. 0) CALL WEL1AL (ISUM, LENX, LCWELL, MXWELL, NWELLS,
1      IUNIT(2), IOUT, IWELCB)
      IF (IUNIT(3) .GT. 0) CALL DRN1AL (ISUM, LENX, LCDRAI, NDRAIN, MXDRN,
1      IUNIT(3), IOUT, IDRNCB)
      IF (IUNIT(8) .GT. 0) CALL RCH1AL (ISUM, LENX, LCIRCH, LCRECH, NRCHOP,
1      NCOL, NROW, IUNIT(8), IOUT, IRCHCB)
      IF (IUNIT(5) .GT. 0) CALL EVT1AL (ISUM, LENX, LCIEVT, LCEVTR, LCEXDP,
1      LCSURF, NCOL, NROW, NEVTOP, IUNIT(5), IOUT, IEVTCB)
      IF (IUNIT(4) .GT. 0) CALL RIV1AL (ISUM, LENX, LCRIVR, MXRIVR, NRIVER,
1      IUNIT(4), IOUT, IRIVCB)
      IF (IUNIT(13) .GT. 0) CALL STR1AL (ISUM, LENX, LCSTRM, ICSTRM, MXSTRM,
1      NSTREM, IUNIT(13), IOUT, ISTCB1, ISTCB2, NSS, NTRIB,
2      NDIV, ICALC, CONST, LCTBAR, LCTTRIB, LCIVAR)
      IF (IUNIT(15) .GT. 0)
1 CALL BRC1AL (LENY, LENYC, LENYL, LINDAT, LIZDAT, LIQDAT, LITQMA, LITQMI,
1 LQMAX, LQMIN, LQSUM, LZQMIN, LZQMAX, LAQMAX, LAQMIN, LA, LZ, LQ, LZP, LQP,
2 LAP, LBP, LRP, LB, LR, LBT, LBTP, LXSTAT, LDX, LT, LRN, LWANGL, LGDATU, LORIE
3 , LBETVE, LSUMET, LSUMCZ, LSCZQS, LSZQET, LITYPO, LZA, LAA, LBB, LBS, LIPT,
4 LQA, LTA, LETA, LFUNET, LROW, LAM, LBMX, LBRNAM, LIJF, LIJT, LNSEC, LXSKT,
5 LPLTBC, LPRTXS, LPRTEC, LPRTSU, LPPLTB, LITYPE, LIBJNC, LNDATA, LIZQBV,
6 LISTAT, LKTTDB, LZQ, LDTT, LDATUM, LZQBVC, LZQPMI, LLARBP, LZHIGH,
7 LZLOW, LLINPR, LARBER, LCLK, LZBOT, MXBH, NSEC, MAXS, MXTDBC,
8 MAXZBD, MAXCZQ, MXWIND, MXJN, NSEG, MXPT, MXMD, MAXQBD, MAXMZQ, ITMUNI,
9 LWINDS, LWINDD, LIDX, LICT, LW, LU, LUU, LBU, LBUU, LZSAV, LQSAV, LZPSAV,
& LQPSAV, IUNIT(15), IOUT, TFCTR, LISTRM, LZPL, LSLKG, IBEGIN, LZN,
& LQLSUM, NELAP, LITRIA, DCFM)
      IF (IUNIT(7) .GT. 0) CALL GHBIAL (ISUM, LENX, LCBNDS, NBOUND, MXBND,
1      IUNIT(7), IOUT, IGHBCB)
      IF (IUNIT(9) .GT. 0) CALL SIPIAL (ISUM, LENX, LCEL, LCFL, LCGL, LCV,
1      LCHDCG, LCLRCH, LCW, MXITER, NPARAM, NCOL, NROW, NLAY,
2      IUNIT(9), IOUT)

```

Table 1. Example listing of a modified main program—Continued

```

      IF (IUNIT(11).GT.0) CALL SORIAL(ISUM,LENX,LCA,LCRES,LCHDCG,LCLRCH,
1      LCIEQP,MXITER,NCOL,NLAY,NSLICE,MBW,IUNIT(11),IOUT)
      IF (IUNIT(20).GT.0) CALL CHDIAL(ISUM,LENX,LCCHDS,NCHDS,MXCHD,
1      IUNIT(20),IOUT)
C
C5-----IF THE "X" ARRAY IS NOT BIG ENOUGH THEN STOP.
      IF (ISUM-1.GT.LENX) STOP
C
C6-----READ AND PREPARE INFORMATION FOR ENTIRE SIMULATION.
      CALL BAS1RP(X(LCIBOU),X(LCHNEW),X(LCSTRT),X(LCHOLD),
1      ISTRT,INBAS,HEADNG,NCOL,NROW,NLAY,NODES,VBVL,X(LCIOFL),
2      IUNIT(12),IHEDFM,IDDNFM,IHEDUN,IDDNUN,IOUT)
      IF (IUNIT(1).GT.0) CALL BCF1RP(X(LCIBOU),X(LCHNEW),X(LCSC1),
1      X(LCHY),X(LCCR),X(LCCC),X(LCCV),X(LCDELRL),
2      X(LCDELCL),X(LCBOT),X(LCTOP),X(LCSC2),X(LCTRPY),
3      IUNIT(1),ISS,NCOL,NROW,NLAY,NODES,IOUT)
      IF (IUNIT(9).GT.0) CALL SIP1RP(NPARM,MXITER,ACCL,HCLOSE,X(LCW),
1      IUNIT(9),IPCALC,IPRSIP,IOUT)
      IF (IUNIT(11).GT.0) CALL SOR1RP(MXITER,ACCL,HCLOSE,IUNIT(11),
1      IPRSOR,IOUT)
      IF (IUNIT(15).GT.0) CALL BRC1RP
& (Y(LINDAT),Y(LIZDAT),Y(LIQDAT),Y(LITQMA),Y(LITQMI),Y(LQMAX),
1 Y(LQMIN),Y(LQSUM),Y(LZQMIN),Y(LZQMAX),Y(LAQMAX),Y(LAQMIN),Y(LA),
2 Y(LZ),Y(LQ),Y(LZP),Y(LQP),Y(LAP),Y(LBP),Y(LRP),Y(LB),Y(LR),
3 Y(LBT),Y(LBTP),Y(LXSTAT),Y(LDX),Y(LT),Y(LRN),Y(LWANGL),Y(LGDATU),
4 Y(LORIEN),Y(LBETVE),Y(LSUMET),Y(LSUMCZ),Y(LSCZQS),Y(LSZQET),
5 YC(LITYPO),Y(LZA),Y(LAA),Y(LBB),Y(LBS),Y(LIPT),Y(LQA),Y(LTA),
6 Y(LETA),Y(LFUNET),YC(LBRNAM),Y(LIJF),
7 Y(LIJT),Y(LNSEC),Y(LXSKT),Y(LPLTBC),Y(LPRTXS),Y(LPRTBC),Y(LPRTSU),
8 Y(LPPLTB),YC(LITYPE),Y(LIBJNC),Y(LNDATA),Y(LIZQBV),Y(LISTAT),
9 Y(LKTTDB),Y(LZQ),Y(LDTT),Y(LDATUM),Y(LZQBVC),Y(LZQPMI),
& Y(LLARBP),Y(LZHGH),Y(LZLOW),Y(LLINPR),Y(LARBER),Y(LCLK),
& Y(LZBOT),MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
& MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,Y(LWINDS),Y(LWINDD),
& Y(LICT),Y(LW),IUNIT(15),IOUT,Y(LISTRM))
C
C7-----SIMULATE EACH STRESS PERIOD.
      DO 300 KPER=1,NPER
      KKPER=KPER
      write(*,*) 'stress period no.',kper
C
C7A-----READ STRESS PERIOD TIMING INFORMATION.
      CALL BAS1ST(NSTP,DELT,TSMULT,PERTIM,KKPER,INBAS,IOUT)
C
C7B-----READ AND PREPARE INFORMATION FOR STRESS PERIOD.
      IF (IUNIT(2).GT.0) CALL WEL1RP(X(LCWELL),NWELLS,MXWELL,IUNIT(2),
1      IOUT)
      IF (IUNIT(3).GT.0) CALL DRN1RP(X(LCDRAI),NDRAIN,MXDRN,IUNIT(3),
1      IOUT)
      IF (IUNIT(8).GT.0) CALL RCH1RP(NRCHOP,X(LCIRCH),X(LCRECH),
1      X(LCDELRL),X(LCDELCL),NROW,NCOL,IUNIT(8),IOUT)
      IF (IUNIT(5).GT.0) CALL EVT1RP(NEVTOP,X(LCIEVT),X(LCEVTR),
1      X(LCEXDP),X(LCSURF),X(LCDELRL),X(LCDELCL),NCOL,NROW,
1      IUNIT(5),IOUT)
      IF (IUNIT(4).GT.0) CALL RIV1RP(X(LCRIVR),NRIVER,MXRIVR,IUNIT(4),
1      IOUT)
      IF (IUNIT(13).GT.0) CALL STR1RP(X(LCSTRM),X(ICSTRM),NSTREM,
1      MXSTRM,IUNIT(13),IOUT,X(LCTBAR),NDIV,NSS,
1      NTRIB,X(LCIVAR),ICALC,IPTFGL)
      IF (IUNIT(7).GT.0) CALL GH1RP(X(LCBNDS),NBOUND,MXBND,IUNIT(7),

```

Table 1. Example listing of a modified main program—Continued

```

1          IOUT)
  IF (IUNIT(20) .GT.0) CALL CHD1RP (X(LCCHDS),NCHDS,MXCHD,X(LCIBOU),
1          NCOL,NROW,NLAY,PERLEN,DELT,NSTP,TSMULT,IUNIT(20),IOUT)
C
C7C-----SIMULATE EACH TIME STEP.
  DO 200 KSTP=1,NSTP
  KKSTP=KSTP
  write(*,*) '    time step no.',kstp
C
C7C1----CALCULATE TIME STEP LENGTH. SET HOLD=HNEW.
  CALL BAS1AD (DELT,TSMULT,TOTIM,PERTIM,X(LCHNEW),X(LCHOLD),KKSTP,
1          NCOL,NROW,NLAY)
  IF (IUNIT(20) .GT.0) CALL CHD1FM (NCHDS,MXCHD,X(LCCHDS),X(LCIBOU),
1  X(LCHNEW),X(LCHOLD),PERLEN,PERTIM,DELT,NCOL,NROW,NLAY)
C
C7C2----ITERATIVELY FORMULATE AND SOLVE THE EQUATIONS.
  DO 100 KITER=1,MXITER
  KKITER=KITER
  write(*,*) '    iteration no.',kiter
C
C7C2A---FORMULATE THE FINITE DIFFERENCE EQUATIONS.
  CALL BAS1FM (X(LCHCOF),X(LCRHS),NODES)
  IF (IUNIT(1) .GT.0) CALL BCF1FM (X(LCHCOF),X(LCRHS),X(LCHOLD),
1          X(LCSC1),X(LCHNEW),X(LCIBOU),X(LCCR),X(LCCC),X(LCCV),
2          X(LCHY),X(LCTRPY),X(LCBOT),X(LCTOP),X(LCSC2),
3          X(LCDELRL),X(LCDELRLC),DELT,ISS,KKITER,KKSTP,KKPER,NCOL,
4          NROW,NLAY,IOUT)
  IF (IUNIT(2) .GT.0) CALL WEL1FM (NWELLS,MXWELL,X(LCRHS),X(LCWELL),
1          X(LCIBOU),NCOL,NROW,NLAY)
  IF (IUNIT(3) .GT.0) CALL DRN1FM (NDRAIN,MXDRN,X(LCDRAI),X(LCHNEW),
1          X(LCHCOF),X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY)
  IF (IUNIT(8) .GT.0) CALL RCH1FM (NRCHOP,X(LCIRCH),X(LCRECH),
1          X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY)
  IF (IUNIT(5) .GT.0) CALL EVT1FM (NEVTOP,X(LCIEVT),X(LCEVTR),
1          X(LCEXDP),X(LCSURF),X(LCRHS),X(LCHCOF),X(LCIBOU),
1          X(LCHNEW),NCOL,NROW,NLAY)
  IF (IUNIT(4) .GT.0) CALL RIV1FM (NRIVER,MXRIVR,X(LCRIVR),X(LCHNEW),
1          X(LCHCOF),X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY)
  IF (IUNIT(13) .GT.0) CALL STR1FM (NSTREM,X(LCSTRM),X(ICSTRM),
1          X(LCHNEW),X(LCHCOF),X(LCRHS),
2          X(LCIBOU),MXSTRM,NCOL,NROW,NLAY,IOUT,NSS,
3          X(LCTBAR),NTRIB,X(LCTTRIB),X(LCIVAR),ICALC,CONST)
  IF (IUNIT(15) .GT.0) CALL BRC1FM
& (Y(LINDAT),Y(LIZDAT),Y(LIQDAT),Y(LITQMA),Y(LITQMI),Y(LQMAX),
1 Y(LQMIN),Y(LQSUM),Y(LZQMIN),Y(LZQMAX),Y(LAQMAX),Y(LAQMIN),Y(LA),
2 Y(LZ),Y(LQ),Y(LZP),Y(LQP),Y(LAP),Y(LBP),Y(LRP),Y(LB),Y(LR),
3 Y(LBT),Y(LBTP),Y(LXSTAT),Y(LDX),Y(LT),Y(LRN),Y(LWANGL),Y(LGDATU),
4 Y(LORIEN),Y(LBETVE),Y(LSUMET),Y(LSUMCZ),Y(LSCZQS),Y(LSZQET),
5 YC(LITYPO),Y(LZA),Y(LAA),Y(LBB),Y(LBS),Y(LIPT),Y(LQA),Y(LTA),
6 Y(LETA),Y(LFUNET),Y(LROW),Y(LAM),Y(LBMX),YC(LBRNAM),Y(LIJF),
7 Y(LIJT),Y(LNSEC),Y(LXSKT),Y(LPLTBC),Y(LPRTXS),Y(LPRTBC),Y(LPRTSU),
8 Y(LPPLTB),YC(LITYPE),Y(LIBJNC),Y(LNDATA),Y(LIZQBV),Y(LISTAT),
9 Y(LKTTDB),Y(LZQ),Y(LDIT),Y(LDATUM),Y(LZQBVC),Y(LZQPMI),
& YL(LLARBP),YL(LZHIG),YL(LZLOW),YL(LLINPR),YL(LARBER),Y(LCLK),
& Y(LZBOT),MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
& MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,Y(LWINDS),Y(LWINDD),
& Y(LIDX),Y(LICT),Y(LW),Y(LU),Y(LUU),Y(LBU),Y(LBUU),Y(LZSAV),
& Y(LQSAV),Y(LZPSAV),Y(LQPSAV),NELAP,IOUT,Y(LZPL),Y(LSLKG),
1          X(LCHNEW),X(LCHOLD),X(LCHCOF),X(LCRHS),
2          X(LCIBOU),NCOL,NROW,NLAY,TFCTR,

```

Table 1. Example listing of a modified main program—Continued

```

4     ISS,NTSAQ,DELT,TOTIM,IBCONV,HCLOSE,Y(LISTRM),IBEGIN,
5     Y(LZN),Y(LQLSUM),Y(LITRIA),DCFM,KKSTP
      IF(IUNIT(7).GT.0) CALL GHBI1FM(NBOUND,MXBND,X(LCBNDS),X(LCHCOF),
1     X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY)
C
C7C2B---MAKE ONE CUT AT AN APPROXIMATE SOLUTION.
      IF(IUNIT(9).GT.0) CALL SI1AP(X(LCHNEW),X(LCIBOU),X(LCCR),X(LCCC),
1     X(LCCV),X(LCHCOF),X(LCRHS),X(LCEL),X(LCFL),X(LCGL),X(LCV),
2     X(LCW),X(LCHDCG),X(LCLRCH),NPARM,KKITER,HCLOSE,ACCL,ICNVG,
3     KKSTP,KKPER,IPCALC,IPRSIP,MXITER,NSTP,NCOL,NROW,NLAY,NODES,
4     IOUT)
      IF(IUNIT(11).GT.0) CALL SOR1AP(X(LCHNEW),X(LCIBOU),X(LCCR),
1     X(LCCC),X(LCCV),X(LCHCOF),X(LCRHS),X(LCA),X(LCRES),X(LCIEQP),
2     X(LCHDCG),X(LCLRCH),KKITER,HCLOSE,ACCL,ICNVG,KKSTP,KKPER,
3     IPRSOR,MXITER,NSTP,NCOL,NROW,NLAY,NSLICE,MBW,IOUT)
C
C7C2C---IF CONVERGENCE CRITERION HAS BEEN MET STOP ITERATING.
C     CHECK CONVERGENCE ON STREAM STAGE TOO.
C     IF(ICNVG.EQ.1) GO TO 110
C     IF(ICNVG.EQ.1) THEN
C     IF(IUNIT(14).GT.0) THEN
C     IF(ISCONV.EQ.1) GO TO 110
C     ELSE
C     GO TO 110
C     END IF
C     IF(IUNIT(15).GT.0) THEN
C     IF(IBCONV.EQ.1) GO TO 110
C     ELSE
C     GO TO 110
C     END IF
C     END IF
100 CONTINUE
      KITER=MXITER
110 CONTINUE
      write(*,*) '          converged at iteration no.',kiter
C
C7C3----DETERMINE WHICH OUTPUT IS NEEDED.
      CALL BAS1OC(NSTP,KKSTP,ICNVG,X(LCIOFL),NLAY,
1     IBUDFL,ICBCFL,IHDDFL,IUNIT(12),IOUT)
C
C7C4----CALCULATE BUDGET TERMS. SAVE CELL-BY-CELL FLOW TERMS.
      MSUM=1
      IF(IUNIT(1).GT.0) CALL BCF1BD(VBNM,VBVL,MSUM,X(LCHNEW),
1     X(LCIBOU),X(LCHOLD),X(LCSC1),X(LCCR),X(LCCC),X(LCCV),
2     X(LCTOP),X(LCSC2),DELT,ISS,NCOL,NROW,NLAY,KKSTP,KKPER,
3     IBCFCB,ICBCFL,X(LCBUFF),IOUT)
      IF(IUNIT(2).GT.0) CALL WEL1BD(NWELLS,MXWELL,VBNM,VBVL,MSUM,
1     X(LCWELL),X(LCIBOU),DELT,NCOL,NROW,NLAY,KKSTP,KKPER,IWELCB,
1     ICBCFL,X(LCBUFF),IOUT)
      IF(IUNIT(3).GT.0) CALL DRN1BD(NDRAIN,MXDRN,VBNM,VBVL,MSUM,
1     X(LCDRAI),DELT,X(LCHNEW),NCOL,NROW,NLAY,X(LCIBOU),KKSTP,
2     KKPER,IDRNCB,ICBCFL,X(LCBUFF),IOUT)
      IF(IUNIT(8).GT.0) CALL RCH1BD(NRCHOP,X(LCIRCH),X(LCRECH),
1     X(LCIBOU),NROW,NCOL,NLAY,DELT,VBVL,VBNM,MSUM,KKSTP,KKPER,
2     IRCHCB,ICBCFL,X(LCBUFF),IOUT)
      IF(IUNIT(5).GT.0) CALL EVT1BD(NEVTOP,X(LCIEVT),X(LCEVTR),
1     X(LCEXDP),X(LCSURF),X(LCIBOU),X(LCHNEW),NCOL,NROW,NLAY,
2     DELT,VBVL,VBNM,MSUM,KKSTP,KKPER,IEVTCB,ICBCFL,X(LCBUFF),IOUT)
      IF(IUNIT(4).GT.0) CALL RIV1BD(NRIVER,MXRIVR,X(LCRIVR),X(LCIBOU),
1     X(LCHNEW),NCOL,NROW,NLAY,DELT,VBVL,VBNM,MSUM,

```

Table 1. Example listing of a modified main program—Continued

```

2      KKSTP, KKPER, IRIVCB, ICBCFL, X(LCIBUFF), IOUT)
      IF(IUNIT(13).GT.0) CALL STR1BD(NSTREM, X(LCSTRM), X(ICSTRM),
1      X(LCIBOU), MXSTRM, X(LCHNEW), NCOL, NROW, NLAY, DELT, VBVL,
2      VBNM, MSUM, KKSTP, KKPER, ISTCB1, ISTCB2, ICBCFL, X(LCIBUFF), IOUT,
3      NTRIB, NSS, X(LCTTRIB), X(LCTBAR), X(LCIVAR), ICALC, CONST, IPTFLG)
      IF(IUNIT(15).GT.0) CALL BRC1BD
& (Y(LINDAT), Y(LIZDAT), Y(LIQDAT), Y(LITQMA), Y(LITQMI), Y(LQMAX),
1 Y(LQMIN), Y(LQSUM), Y(LZQMIN), Y(LZQMAX), Y(LAQMAX), Y(LAQMIN), Y(LA),
2 Y(LZ), Y(LQ), Y(LZP), Y(LQP), Y(LAP), Y(LBP), Y(LRP), Y(LB), Y(LR),
3 Y(LBT), Y(LBTP), Y(LXSTAT), Y(LDX), Y(LT), Y(LRN), Y(LWANGL), Y(LGDATU),
4 Y(LORIEN), Y(LBETVE), Y(LSUMET), Y(LSUMCZ), Y(LSCZQS), Y(LSZQET),
5 YC(LITYPO), Y(LZA), Y(LAA), Y(LBB), Y(LBS), Y(LIPT), Y(LQA), Y(LTA),
6 Y(LETA), Y(LFUNET), Y(LROW), Y(LAM), Y(LBMX), YC(LBRNAM), Y(LIJF),
7Y(LIJT), Y(LNSEC), Y(LXSKT), Y(LPLTBC), Y(LPRTXS), Y(LPRTBC), Y(LPRTSU),
8Y(LPPLTB), YC(LITYPE), Y(LIBJNC), Y(LNDATA), Y(LIZQBV), Y(LISTAT),
9Y(LKTTDB), Y(LZQ), Y(LDTT), Y(LDATUM), Y(LZQBVC), Y(LZQPMI),
&YL(LLARBF), YL(LZHGH), YL(LZLOW), YL(LLINPR), YL(LARBER), Y(LCLK),
& Y(LZBOT), MXBH, MXJN, MAXS, MXPT, MXTDBC, MXMD, MAXZBD,
& MAXQBD, MAXCZQ, MAXMZQ, MXWIND, MAXBD, MXOTDT, Y(LWINDS), Y(LWINDD),
& Y(LIDX), Y(LICT), Y(LW), Y(LU), Y(LJU), Y(LBU), Y(LBUU), Y(LZSAV),
& Y(LQSAV), Y(LZPSAV), Y(LQPSAV), NELAP, IOUT, Y(LZPL), Y(LSLKG),
1      X(LCHNEW), X(LCHOLD),
2      X(LCIBOU), NCOL, NROW, NLAY, TFCTR,
4      NNTSAQ, DELT, KKSTP, KKPER, Y(LISTRM),
1 VBVL, VBNM, MSUM, IMPCB1, ICBCFL, IHDDFL, X(LCIBUFF), IPTFL2,
2 Y(LZN), ISS, Y(LQLSUM), Y(LITRIA), DCFM)
      IF(IUNIT(7).GT.0) CALL GH1BD(NBOUND, MXBND, VBNM, VBVL, MSUM,
1      X(LCBNDS), DELT, X(LCHNEW), NCOL, NROW, NLAY, X(LCIBOU), KKSTP,
2      KKPER, IGHBCB, ICBCFL, X(LCIBUFF), IOUT)
C
C7C5---PRINT AND OR SAVE HEADS AND DRAWDOWNS. PRINT OVERALL BUDGET.
      CALL BAS1OT(X(LCHNEW), X(LCSTRT), ISTRT, X(LCIBUFF), X(LCIOFL),
1      MSUM, X(LCIBOU), VBNM, VBVL, KKSTP, KKPER, DELT,
2      PERTIM, TOTIM, ITMUNI, NCOL, NROW, NLAY, ICNVG,
3      IHDDFL, IBUDFL, IHEDFM, IHEDUN, IDDNFM, IDDNUN, IOUT)
C
C7C6----IF ITERATION FAILED TO CONVERGE THEN STOP.
      IF(ICNVG.EQ.0) STOP
      200 CONTINUE
      300 CONTINUE
C
C8-----END PROGRAM
      STOP
C
      END

```

9 Y(LKTTDB),Y(LZQ),Y(LDTT),Y(LDATUM),
Y(LZQBVC),Y(LZQPMI),
1 YL(LLARBP),YL(LZHIG),YL(LZLOW),YL
(LLINPR),YL(LARBER),Y(LCLK),
2 Y(LZBOT),MXBH,MXJN,MAXS,MXPT,MXT-
DBC,MXMD,MAXZBD,
3 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,
MAXBD,MXOTDT,Y(LWINDS),Y(LWINDD),
4 Y(LIDX),Y(LICT),Y(LW),Y(LU),Y(LUU),
Y(LBU),Y(LBUU),Y(LZSAV),
5 Y(LQSAV),Y(LZPSAV),Y(LQPSAV),NELAP,
IOUT,Y(LZPL),Y(LSLKG),
6 X(LCHNEW),X(LCHOLD),X(LCHCOF),X(LC-
RHS),
7 X(LCIBOU),NCOL,NROW,NLAY,TFCTR,
8 ISS,NTSAQ,DELT,TOTIM,IBCONV,HCLOSE,
Y(LISTRM),IBEGIN,
9 Y(LZN),Y(LQLSUM),Y(LITRIA),DCFM,KKS-
TP)

Between comments C7C4 and C7C5 is added:

```
IF(IUNIT(??).GT.0) CALL BRC1BD
& (Y(LINDAT),Y(LIZDAT),Y(LIQDAT),Y(LITQ-
MA),Y(LITQMI),Y(LQMAX),
1 Y(LQMIN),Y(LQSUM),Y(LZQMIN),Y(LZQ-
MAX),Y(LAQMAX),Y(LAQMIN),Y(LA),
2 Y(LZ),Y(LQ),Y(LZP),Y(LQP),Y(LAP),Y(LBP),
Y(LRP),Y(LB),Y(LR),
3 Y(LBT),Y(LBTP),Y(LXSTAT),Y(LDX),Y(LT),
Y(LRN),Y(LWANGL),Y(LGDA TU),
4 Y(LORIEN),Y(LBETVE),Y(LSUMET),Y(LSU-
MCZ),Y(LSCZQS),Y(LSZQET),
5 YC(LITYPO),Y(LZA),Y(LAA),Y(LBB),Y(LBS),
Y(LIPT),Y(LQA),Y(LTA),
6 Y(LETA),Y(LFUNET),Y(LROW),Y(LAM),Y
(LBMX),YC(LBRNAM),Y(LIJF),
7 Y(LIJT),Y(LNSEC),Y(LXSKT),Y(LPLTBC),
Y(LPRTXS),Y(LPRTBC),Y(LPRTSU),
8 Y(LPPLTB),YC(LITYPE),Y(LIBJNC),Y(LN-
DATA),Y(LIZQBVC),Y(LISTAT),
9 Y(LKTTDB),Y(LZQ),Y(LDTT),Y(LDATUM),
Y(LZQBVC),Y(LZQPMI),
1 YL(LLARBP),YL(LZHIG),YL(LZLOW),YLL-
LINPR),YL(LARBER),Y(LCLK),
2 Y(LZBOT),MXBH,MXJN,MAXS,MXPT,MXT-
DBC,MXMD,MAXZBD,
3 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,
MAXBD,MXOTDT,Y(LWINDS),Y(LWINDD),
4 Y(LIDX),Y(LICT),Y(LW),Y(LU),Y(LUU),
Y(LBU),Y(LBUU),Y(LZSAV),
5 Y(LQSAV),Y(LZPSAV),Y(LQPSAV),NELAP,
IOUT,Y(LZPL),Y(LSLKG),
6 X(LCHNEW),X(LCHOLD),
7 X(LCIBOU),NCOL,NROW,NLAY,TFCTR,
```

8 NTSAQ,DELT,KKSTP,KKPER,Y(LISTRM),
9 VBV,VBNM,MSUM,IMPCB1,ICBCFL,IHD-
DFL,X(LCBUFF),IPTFL2,
1 Y(LZN),ISS,Y(LQLSUM),Y(LITRIA),DCFM)

The user must specify a number between 13 and 24 for IUNIT(??), which is defined in the basic package (McDonald and Harbaugh, 1988, p. 4-9 to 4-12). In the example main program list, IUNIT(15) is used.

DATA ENTRY

The format for entering data into MODFLOW and BRANCH' when coupled by MODBRANCH is nearly the same as using each model separately. The original input instructions were described by McDonald and Harbaugh (1988) and Schaffranek and others (1981). Modifications to these original formats are described below.

Modifications in MODFLOW Data Input—The only difference in MODFLOW input is the instruction to use the MODBRANCH code. This occurs in the input to the basic package. Element 15 in the IUNIT array specifies the FORTRAN unit number of the input data file for MODBRANCH. If IUNIT(15) is set to zero, MODBRANCH is not used.

Modifications in BRANCH' Data Input—Although most of the input data for BRANCH' are the same, there are several added items. At the beginning of the BRANCH' data set, two records are added containing input data for the BRC1AL routine. The records are

Data	-	MXBH	NSEC	MAXS	MXTDBC	MAXZBD	MAXCZQ	MXWIND
Format	-	110	110	110	110	110	110	110
Data	-	MXJN	NSEG	MXPT	MXMD	MAXQBD	MAXMZQ	DCFM
Format	-	110	110	110	110	110	110	F103

The record variables are

MXBH - Maximum number of branches in the network.
NSEC - Maximum cross sections per branch.
MAXS - Maximum number of cross sections in the entire channel network.
MXTDBC - Maximum number of boundaries in the network.
MAXZBD - Maximum number of stage boundary value data held in storage for computation.
MAXCZQ - Maximum number of daily computed results held in storage for plotting purposes.
MXWIND - Maximum number of wind data points input.
MXJN - Maximum number of junctions in the network.
NSEG - Maximum segments per branch.
MXPT - Maximum number of points used to define a cross section.
MXMD - Maximum number of measured data locations accommodated in the network.
MAXQBD - Maximum number of discharge boundary value data held in storage for computation.
MAXMZQ - Maximum number of measured data held in storage for plotting purposes.

DCFM - Maximum value of the multiplier for the friction term in the momentum equation when the channel is dry.

All of these variables, except for DCFM, are BRANCH array dimensions described by Schaffranek and others (1981). DCFM is described previously in this text.

Another modification to the BRANCH' input data occurs in the second initial condition record for cross section (Schaffranek and others, 1981). The new input format is

Data -	ORIENT	BETVEL	ISTRM	ISTRM
	(MAXS)	(MAXS)	(1, MAXS)	(2, MAXS),
Format -	F10.3	F10.3	I10	I10
Data -	ISTRM	CLK	ZBOT	
	(3, MAXS)	(MAXS)	(MAXS)	
Format -	I10	F10.4	F10.4	

The new variables added to these records are

- ISTRM (1, MAXS) - Layer number of aquifer model cell containing river reach.
- ISTRM (2, MAXS) - Row number of aquifer model cell containing river reach.
- ISTRM (3, MAXS) - Column number of aquifer model cell containing river reach.
- CLK (MAXS) - Leakage coefficient for reach.
- ZBOT (MAXS) - Elevation of riverbed.

To accommodate the additional cross-section data points added by the program to account for drying, the maximum number of cross-sectional data points per cross section (MXPT) should be set at least four more than the actual number of points input by the user.

MODULE BRC1AL

This module allocates space in three large arrays in the main program of MODFLOW for storing the arrays used in BRANCH'. The numbered sections in the narrative correspond to the numbered steps in the flowchart for BRC1AL shown in figure 6.

NARRATIVE

1. Print a message identifying the package. The array dimensions MXBH, NSEC, MAXS, MXTDBC, MAXZBD, MAXCZQ, MXWIND, MXJN, NSEG, MXPT, MXMD, MAXQBD, and MAXMZQ are entered along with a DCFM value. The multiplier DCFM is the factor by which the friction term in the momentum equation is multiplied when the channel is dry. The data entry format is described in the data entry section. The definition of the array dimensions can be found in the report by Schaffranek and others (1981) and in the data entry section.

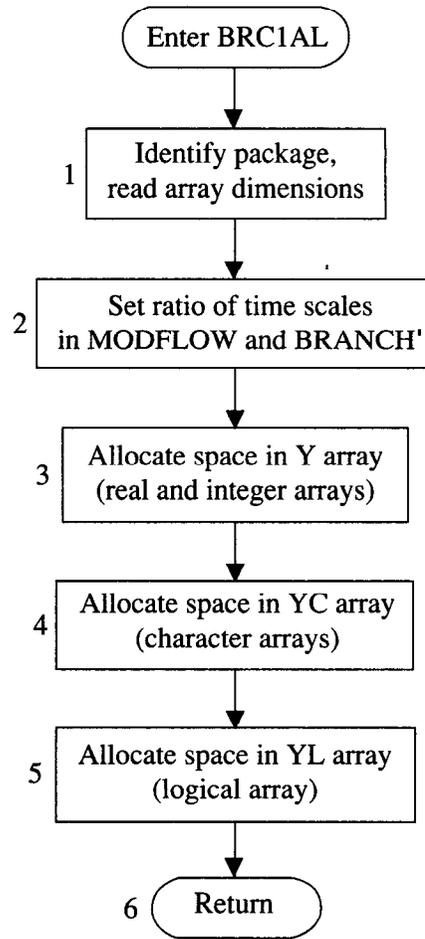


Figure 6. Flowchart of allocation module (BRC1AL).

2. Set the ratio of time scales in MODFLOW and BRANCH'. Although BRANCH' always works in seconds, MODFLOW can work in seconds, minutes, hours, days, and years, or the time units can be undefined. This option, defined as the variable ITMUNI (McDonald and Harbaugh, 1988), is used to calculate the ratio of the time units in the models, the variable TFCTR. An error message is printed if the undefined option is selected in MODFLOW.
3. Allocate space in the Y array. All BRANCH' arrays that contain real or integer numbers are stored in the Y array, with positions and space allocated by this routine.
4. Allocate space in the YC array. All BRANCH' arrays that contain character variables are allocated positions and space in the YC array.
5. Allocate space in the YL array. All BRANCH' arrays that contain logical variables are allocated positions and space in the YL array.
6. Return to MODFLOW.